komputer 7

lipiec 1988

popularny miesięcznik informatyczny: nr 7(28) '88 180 zł.

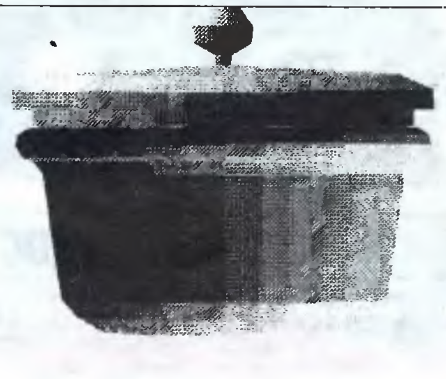


Polskie litery dla Atari ST

Nieśmiertelny robal

Dostęp...

Menu



Szef kuchni poleca

W okresie wakacji numery pisma powinny być nieco lżejsze, bo na plaży czy pomoście nad jeziorem lektura trudnych tekstów jest niezwykle męcząca, a nie wypada utrudniać wypoczynku Czytelnikom i redakcyjnym kolegom (ktoś musi wypoczywać, aby pracować mógł ktoś). Proponujemy więcej niż zazwyczaj tekstów publicystycznych. Materiały zamieszczane w części sygnowanej "Kurier" czyta się łatwiej niż te, które znalazły się pod winietami "W domu" czy "W pracy", ale - w tym numerze - ich ciężar gatunkowy jest nie mniejszy. Wprost przeciwnie. Codzienne życiowe problemy ludzi niepełnosprawnych nie tylko nie są przez społeczeństwo dostrzegane, ale już samo podjęcie tego tematu uchodzi za rzecz wstydliwą, o której w towarzystwie się nie rozmawia. A już w żadnym wypadku nie należy o tym mówić w obecności osoby niepełnosprawnej, "bo ona jest taka biedna". Tyle tylko, że litość rzadko bywa uczuciem potrzebnym, a z całą pewnością na nic się nie przyda inwalidom i chorym. Tymczasem możliwości skutecznej pomocy jest wiele, szczególnie duże perspektywy daje wykorzystanie komputerów i niektóre zastosowania pokazujemy w tym numerze "Komputera".

O tym jak wiele można i trzeba zrobić w dziedzinie edukacji, traktuje dyskusja redakcyjna, której tytuł znakomicie oddaje istotę problemu ludzi niepełnosprawnych: "Wyrwać się z getta kalectwa". W reportażu "Dostęp" znajdziemy przykłady niemożności, ale i szanse spełnienia zawodowych ambicji. Cennym materiałem wydają się rozważania pod tytułem "Komputery pomogą niewidomym?!", także dlatego, że ich autorem jest niewidomy informatyk, osoba, która najlepiej potrafi ocenić potrzeby, a zarazem wie na co można liczyć w naszej polskiej rzeczywistości. Uzupełnieniem bloku jest sprawozdanie z Międzynarodowego Salonu Medycznego "Salmed'88". Rozmowa z wiceprezydentem europejskiego oddziału firmy Autodesk winna usatysfakcjonować tych wszystkich, którzy szukają bliższych danych o programach typu AutoCAD ("AutoCAD w Polsce").

Miłośnicy komputera "W domu" poza stałymi rubrykami ("Poke n,∞", "Klub Mistrzów Komputera", "Przegląd najnowszych gier", "Mikroprogramy dla Atari XE/XL") znajdą w tym numerze kolejny odcinek cyklu poświęconego systemowi operacyjnemu CP/M, a także opis możliwości nowego programu Petera Nortona, znakomicie ułatwiającego wykorzystywanie wszelkich "sztuczek i chwytów" wcześniej przezeń napisanych ("Norton Integrator"). Wróciliśmy raz jeszcze do algorytmów sortowania oraz edytorów tekstu; tym razem przedstawiamy Atariwriter Plus przeznaczony dla modelu 130 XE. "Polskie znaki dla Atari ST" to propozycja dla użytkowników tych maszyn. Inne rozwiązania opublikujemy w następnych numerach.

Lipcowy wypoczynek uwzględnili w swoich planach nawet poważni panowie pasjonujący się komputerem "W pracy". Tym razem proponują twórczą zabawę polegającą na tworzeniu własnych, nowych światów. Powstają one z "automatów komórkowych", których ideę autorzy przedstawiają ("Nieśmiertelny robal"). W konwencji poważnej zabawy utrzymany jest również tekst "Zabawy z ekranem", z którego można się dowiedzieć, w jaki sposób wykorzystać niedostępną wprost pamięć ekranu w komputerach klasy IBM PC. Coraz powszechniejsze w zastosowaniach profesjonalnych staje się wykorzystanie obrazu nie tylko do ilustracji danych, ale jako podstawy do ich otrzymywania. Niezbędne są wówczas urządzenia umożliwiające cyfrowe przetwarzanie obrazu. O jednym z nich piszemy w tekście "Frame Grabber". Wakacje to również okres wzmożonych zakupów komputerów i ich licznych kopii oferowanych przez zachodnie firmy. Jedną z nich (kopii nie firm) testowaliśmy, a wyniki prób omawiamy na łamach ("Kolgar portable AT").

Życzę przyjemnego urlopu i miłej lektury. Z "Komputerem" na plażę? A czemuż by nie?

Stanisław Marek Królak

7 (28)

Popularny Miesięcznik Informatyczny – pismo miłośników i użytkowników mikrokomputerów redagują:

Marek Młynarski (red. nacz.)
Grzegorz Eider (z-ca red. nacz.)
Władysław Majewski (z-ca red. nacz.)
Stanisław M. Królak (sekr. red.)
Marek Car (publicystyka)
Grzegorz Czapkiewicz (programy)
Mariusz Dec (sprzęt)
Zenon Rudak (sprzęt)
Tomasz Zieliński (listy)
oraz współpracownicy:
Zbigniew Blewoński, Rafał Brzeski, Andrzej Kadlof, Jarosław Kania, Zbigniew Kasprzycki, Jacek A. Likowski, Tomasz Mazur, Wiesław Migut, Jarosław Młodzki, Adam Nowicki, Wojciech Olejniczak, Sergiusz Piotrowski, Juliusz Rawicz, Leszek Rudak, Jakub Tatarkiewicz, Roland Waclawek (Katowice), Tadeusz Wilczek, Andrzej Załuski (Kraków).

Redakcja graficzno-techniczna:
Stefan Szczypka (kier.)
Małgorzata Luzińska
Piotr Kakiet
Magdalena Stachorzyńska (operatorka komputera)

Redakcja programów komputerowych:
Jerzy Pusiak – kier.
Leszek Gołębiowski
Krzysztof Matey
ul Koszykowa 6A
00-564 Warszawa
282201 w. 312

Korekta: Maria Omiecińska,
Romualda Miarecka
Sekretariat: Izabela Radzikowska

Wydawca: Warszawskie Wydawnictwo Prasowe RSW „Prasa-Książka-Ruch", Al. Jerozolimskie 125/127, 02-017 Warszawa, tel. centrali: 28-52-31. Redakcja: ul. Koszykowa 6A, 00-564 Warszawa, tel. 28-22-01 w. 243 lub 290 telex 813230 csdk pl
Skład i druk: Prasowe Zakłady Graficzne, Łódź, ul. Armii Czerwonej 28.
Cena: 180 zł. Zam. 1643/88, U-23.

Prenumeratę od instytucji przyjmują oddziały RSW, a od osób prywatnych poczta (na wsi także doręczyciele). Cena prenumeraty rocznej 2160 zł, półrocznej 1080 zł, kwartalnej 540 zł. Prenumeratę ze zleceniem wysyłki za granicę (droższą o 50% dla osób prywatnych i o 100% dla instytucji) przyjmuje Centrala Kolportażu RSW, ul. Towarowa 28, 00-958 Warszawa, I PKO BP XV Oddz. W-wa 1658-201045-139-11.

Ogłoszenia przyjmuje w redakcji akwizytor: Krzysztof Karpiński tel. 28-22-01 w. 243 lub 290 oraz Biuro Reklamy Prasowej i Ogłoszeń, ul. Poznańska 38, 00-689 Warszawa, tel. 28-23-09. Zamawiając ogłoszenia listownie należy podać datę i miejsce wpłaty (konto W.W.P: NBP III O/M Warszawa nr 1036-6969 z zaznaczeniem „ogłoszenie w KOMPUTERZE").

1cm² ogłoszenia kosztuje 600 zł, najmniejsze ogłoszenie – 15 cm², kolor – 100% drożej. 1 cm² ogłoszenia na kolumnie ekspresowej – 1200 zł. Za treść ogłoszeń redakcja nie odpowiada.

Nakład 115 000 egz.
Nr indeksu 36-345 ISSN 0860-2514

\Kurier\

Marek Młynarski

Ewolucja

Sezon wakacyjno-urlopowy osłabił nieco ruch na komputerowym rynku. Wprawdzie, jak uczy doświadczenie, może się jeszcze wszystko zdarzyć, wszystko może zostać usankcjonowane odpowiednim aktem prawnym, to jednak w chwili pisania tego tekstu (koniec maja) pozwalam sobie założyć, że nic głupiego nie nastąpi. Mam tu na myśli coraz głośniej precyzowane ciągoty do położenia kresu prywatnemu importowi komputerów przez wprowadzenie "odpowiednich" ceł.

Doprawdy nie tylko ręce i uszy opadają, gdy słyszy się podobne propozycje formułowane przez (wydawałoby się) poważnych ludzi. Jak wiadomo, komputery same z siebie nie chronią przed idiotyzmami, ale żeby je prowokowały...! Na pustyni naszej gospodarki popędzanej naprzód niestety nie prawami ekonomicznymi, a specjalnymi uprawnieniami dla jej popychaczy (sądzę, że nie tylko rząd boleje nad tym kuriozum), zamyśla się zatkać jeden z ostatnich kraników, z których jako tako ciecze woda. Słowa poety (współczesnego, ale już niemal klasyka) - "co by tu jeszcze spieprzyć, panowie?... co by tu jeszcze?" mają, jak się okazuje, straszną moc.

Załóżmy jednak ryzykowne stwierdzenie, że nic złego się nie stanie i wkrótce napłynie do Polski nowa fala komputerów nabytych podczas letnich zachodnio-zagranicznych wojaży. Zwiększy się grono użytkowników komputerów, a co za tym idzie, wśród nabywców pism komputerowych znajdzie się kolejna grupa rozpoczynających od zera swą edukację. Nasze pisma uległy tymczasem ewolucji, poszły naprzód, w minionych numerach omówiły już podstawowe informacje o komputerach i najpopularniejszych programach. Normalnym trybem rzeczy informacje takie powinny w poprawionej i uzupełnionej wersji ukazać się w wydaniach książkowych. I częściowo tak się dzieje, ale straszliwe fatum w postaci cyklu wydawniczego paraliżuje część wysiłków wydawnictw książkowych.

Co więc mają czynić dziesiątki tysięcy posiadaczy najpopularniejszych komputerów (i tylko komputerów), skoro nawet przeznaczony dla początkujących "Bajtek" jeden ze swych ostatnich numerów poświęca kwestii DTP (lansujemy polską nazwę - pulpit wydawniczy), które to programy naprawdę niewiele znaczą bez drukarki (nie wspominając już o kwestii typu stosowanego komputera). No, a wiadomo, drukarka to ...set tysięcy zł.

W moim przekonaniu obecnie w stosunkowo najcięższej sytuacji są posiadacze "małego" Atari, książki dopiero są w produkcji, a artykuły prasowe niezbyt liczne. Jeszcze gorzej mają korzystający z Commodore. Z kolei użytkownicy Spectrum napotykają trudności innego rodzaju - napisano już tyle, że bardzo trudno o coś nowego, a przecież jest to komputer trzymający się mocno. Ten skrót objawów choroby dotyczy jedynie komputerów najpopularniejszych na rynku.

Teraz próba diagnozy. Przez kilka lat utworzyła się spora grupa osób interesujących się "małą" informatyką, chłonąca wszelkie wieści na ten temat, niestety pozbawiona możliwości praktycznego sprawdzania swych umiejętności. Objawieniem stało się kilka lat temu Spectrum, w stosunkowo krótkim czasie całkowicie "rozgryzione". Ludzie ćwiczący "na sucho" otrzymali wreszcie narzędzie. Fascynacja trwałaby pewnie jeszcze do dziś, pomimo poznania każdego milimetra wnętrza Spectrum, gdyby nie uparty postęp techniczny, gdzieniegdzie wyraźniej niż u nas widoczny. Na horyzoncie pojawiły się skośnookie kopie IBM, a ich cena nie stanowiła zapory nie do obalenia. Poczciwe Spectrum powędrowało do szafy. Ci, którzy używali go jako swego pierwszego, umiłowanego komputera, na swym biurku postawili komputery 16-bitowe.

Nastąpiła niejako naturalna ewolucja i dotyczy ona zarówno prywatnych jak "instytucjonalnych" użytkowników. Wprowadzone na nasz rynek nieco później "małe" Atari czeka na tych, którzy - podobnie jak prof. Tadeusiewicz - widzą w nim nie tylko maszynkę do gier, ale i sprawne, o dużych możliwościach narzędzie. Dotyczy to także innych typów komputerów. Z prawdziwą więc przyjemnością publikujemy artykuły oraz "sztuczki i chwyty" przysyłane przez ich użytkowników do naszej redakcji.

Czas na lekarstwo. W najbliższych numerach "Komputera" utworzymy stałą rubrykę, w której będziemy powracali do zagadnień podstawowych, w tym i omawianych już na naszych łamach. Nie mamy jednak zamiaru zamieszczać przedruków - nawet przy powrocie do już publikowanych materiałów będą to nowe, uzupełnione opracowania, wzbogacone o nasze doświadczenia i informacje zebrane już po pierwszej publikacji. Powrócimy więc do informacji interesujących zarówno zaawansowanych jak i początkujących użytkowników najpopularniejszych w Polsce mikrokomputerów, których - pomimo masowego wyposażania biur i instytucji w 16-bitowce - wciąż jest najwięcej.

Może nie będzie to dziś specjalną pociechą dla najliczniejszej grupy korzystającej z komputerów 8-bitowych, ale jestem przekonany, że za trzy-cztery lata zaszczytną nazwą "złomu i żelastwa" będą obdarzane tak dziś pożądane kopie IBM-XT i AT. Być może tego losu unikną inne komputery, dziś wyprzedzające ogólnoświatowy standard, (piszemy o nich regularnie), ale oznacza to jedynie odłożenie "wyroku" o rok lub dwa. Niestety nie potrafię dziś podać recepty na to, wydaje się, nieuchronne schorzenie.

\Kurier\

Władysław Majewski

Branża

W ciągu ostatnich dwóch lat powstało w Polsce kilkaset - lub jak sądzą niektórzy - nawet kilka tysięcy nowych firm "komputerowych". W większości są to jednostki gospodarcze skromne kadrowo - liczące od kilku do kilkudziesięciu pracowników i współpracowników, zajmujące się importem lub skupem sprzętu komputerowego. jego serwisem oraz instalacją u klientów, produkcją oraz wdrażaniem oprogramowania lub związanymi ze środkami informatyki usługami.

Zakres działania nowej branży nie jest określony zbyt ostro - wiele z działających w tym środowisku firm zajmuje się także doradztwem organizacyjnym, wdrażaniem nowej technologii nie związanej z informatyką lub bardzo specjalnymi jej zastosowaniami - w instalacjach pomiarowych, medycynie, projektowaniu, poligrafii. Mieszczą się w niej jednostki nowo powstałe i takie jak Składnica Harcerska lub Bomis, które dopiero w ostatnim czasie rozszerzyły swe zainteresowania na środki informatyki.

Gromadzi ona jednostki spółdzielcze, państwowe, rzemieślnicze i spółki, firmy jednoosobowe i takie z miliardowym obrotem. Czy mają one coś ze sobą wspólnego, czy powinny współpracować i dysponować wspólną reprezentacją?

Wydaje się, że głównym wyróżnikiem branży jest znaczenie kapitału ludzkiego. Nawet firmy o wielomiliardowych obrotach nie mają w niej zbyt dużego majątku trwałego, a własnych murów często nie mają w ogóle.

O sukcesach nie decydują w tej branży pieniądze ani dostęp do zaopatrzenia. Gwałtowne upadki i oszałamiające kariery wiążą się z odejściem lub skaperowaniem solidnych fachowców, opłacanych często lepiej niż w klubach piłkarskich.

Z wielu względów - także w wyniku formalnych ograniczeń - tego rodzaju kapitał trudno skoncentrować w jednym miejscu. Czy oznacza to jednak konieczność rezygnacji z korzyści ekonomicznych, jakie daje wielka skala działania? Niekoniecznie. Koncentracja własności lub zarządzania jest tylko pozornie najprostszą drogą ku wielkiej skali działania. Równie efektywna może być współpraca i specjalizacja. Rozwój w tym kierunku blokowany jest częściowo przez obowiązujące zasady poboru podatku obrotowego, a po części przez brak właściwych form organizacyjnych - firmy komputerowe sztucznie przydzielane są do różnych cechów, zrzeszeń lub po prostu nigdzie.

Brak forum, pozwalającego kształtować współpracę, wyrażać publicznie interesy środowiska (choćby potrzebę rozwoju infrastruktury telekomunikacyjnej i właściwego honorowania wiedzy czy ochrony własności intelektualnej), planować wspólne inwestycje. Wiele ostatnich inicjatyw wskazuje jednak, że potrzeba takiego forum jest dziś powszechnie dostrzegana i akceptowana.

Mam nadzieję, że inicjatywy te szybko wyjdą z fazy planów i że zostaną wsparte przez odpowiednie urzędy. Poza wszelkimi względami merytorycznymi powinny za tym przemawiać także doraźne względy polityczne: branża komputerowa jest dziś dużym i silnym środowiskiem inteligencji najżywotniej zainteresowanym w stabilizacji politycznej kraju i szybkich postępach reformy.

Na 10 dni przed drukiem

Wkrótce:

* III Międzynarodowa Szkoła "Mikrokomputer'88 - Projektowanie, Praktyka, Nauczanie", Bierutowice, 20-23 wrzesień 1988.
* IV Ogólnokrajowa konferencja "Komputery w nauczaniu", Wałbrzych, 28 - 30 wrzesień 1988 r., organizowana przez Radę Wojew. NOT.
* SOFT-TARG '88, Ośrodek Postępu Technicznego, Chorzów, IX.88
* INFORMACJA '88, Hala "Spodek", Katowice, 14-17.11.88 wraz z towarzyszącymi seminariami "Informatyka w medycynie" i "Desktop Publishing" (prowadzone przez naszą redakcję). Organizator: PRO-INFO, Katowice (Uwaga: zmiana terminu imprezy!!)
* INFOGRYF '88 - IX Kołobrzeskie Dni Informatyki wraz z giełdą oprogramowania i konferencją "Informatyka w przedsiębiorstwie", organizowane przez Ośrodek Postępu Organizacyjnego TNOiK, Oddział w Szczecinie, tel. 428-76, 447-25, ul. Pocztowa 30/12, 70-360 Szczecin - 18-21. październik 88 r.

40 lat polskiej informatyki

W grudniu 1948 r. rozpoczęła pracę grupa Aparatów Matematycznych w Instytucie Matematycznym. Rok 1988 jest więc 40 rokiem prac badawczych, wdrożeniowych i produkcyjnych w dziedzinie informatyki w Polsce.

Polskie Towarzystwo Informatyczne wraz ze Stowarzyszeniem Elektryków Polskich organizuje 22 października 1988 r. konferencję naukową poświęconą pierwszym 20 latom informatyki w Polsce.

Zainteresowani udziałem w konferencji i jej przygotowaniu proszeni są o kontakt z PTI, ul. Rzymowskiego 30, 02-697 Warszawa, t 43 74 44.

Mikrolaur' 89

W bieżącym roku Klub Użytkowników Mikrokomputerów Profesjonalnych, Polskie Towarzystwo Informatyczne, PRiWHZ Agpol oraz nasza redakcja po raz drugi organizują konkurs "Mikrolaur".

Do udziału w nim zapraszamy autorów (osoby prawne i twórcy indywidualni) nowoczesnych i wdrożonych już rozwiązań sprzętowych i programowych oraz - w osobnej kategorii - zagranicznych wystawców na IV Międzynarodowej Wystawie "Computer'89".

Prace krajowe prosimy zgłaszać do 30 listopada 1988 r. na adres KUMP, Warszawa, ul. Dantyszka 12. Wystawcy zagraniczni mogą zgłaszać eksponaty do dnia otwarcia wystawy "Computer'89".

Sąd konkursowy pod przewodnictwem prof. Michała Kleibera przyzna medale i dyplomy "Mikrolaur'89". Ich rozdanie nastąpi podczas IV Międzynarodowej Wystawy "Computer'89".

AutoCAD dla tropiciela...

Firma Autodesk Ltd. za każdy dowód sprzedania w Polsce pirackiej kopii najnowszej, 9 wersji programu AutoCAD oferuje bezpłatnie wartą 2500 funtów licencję na użytkowanie tego programu!

... a kurs dla twórcy

Autoryzowane Centrum Szkolenia użytkowników pakietu AutoCAD (firma APLICOM, tel. 57 20 94, Łódź) zaprasza do udziału w konkursie dla autorów aplikacji tego pakietu - jego adaptacji do określonej grupy prac konstrukcyjno-projektowych, procedur lub programów w AutoLISP, bibliotek typowych elementów, intersujących rysunków i innych wartych upowszechnienia rozwiązań.

Nagrodami są trzy miejsca na laboratoryjnym kursie szkoleniowym dla zaawansowanych, profesjonalna literatura dotycząca AutoCAD-a, demonstracyjna kopia najnowszej, 9 wersji programu. Termin konkursu upływa 20 lutego 1989 r., ale najciekawsze z prac, nadesłanych do końca września będą prezentowane na wystawie organizowanej przez Autodesk Ltd. w Moskwie w październiku br.

Konkurs na dowcip...

o komputerach, ogłoszony przez nas wspólnie z firmą Videx, dobiega końca - termin zgłaszania propozycji upływa 30 września br.

Przypominamy (pełna informacja ukazała się w numerze 5/88), że nagrodą w tym konkursie jest komputer Atari XE.

Goście z Warmii

13 lipca odwiedziła nas grupa młodzieży z obozu dla wybitnie uzdolnionej młodzieży prowadzonego przez p. Jana Szachowicza z Lidzbarka Warmińskiego. Ich obozowa komputerowa gazetka wzbudziła nasze szczere uznanie. Gratulujemy!

NEAT AT

Dynamicznie ostatnio zdobywająca rynek firma Chips Technology opracowała komplet 5 kości, pozwalających zbudować komputer zgodny z IBM AT pracujący z zegarem 20 MHz. Tego typu komputery natychmiast znalazły się w ofercie licznych firm dalekowschodnich z cenami znacznie niższymi od cen równie szybkich komputerów klasy 386.

Przepraszamy...

Czytelników oraz Adama Nowickiego, autora cyklu "Praktyka programowania" za pomylenie kolejności jego odcinków. Odcinek drugi ukazał się w numerze 6/88, trzeci ukaże się w numerze 8/88, a pierwszy, wprowadzający, opublikujemy dopiero w numerze 10/88.

Informujemy równocześnie, że drugi odcinek tekstu Wiktora Daszczuka o programowaniu współbieżnym, którego pierwsza część opublikowana została w numerze 4/88 ukaże się również dopiero w numerze 10/88.

Japończycy w opałach ???

Komisja EWG dla ochrony rynku Wspólnoty przed napływem podejrzanie tanich wyrobów zagranicznych zajęła się ofertą firm japońskich.

Producenci towarów tanich i dobrych podejrzewani są o sprzedawanie ich po cenach niższych od kosztów wytwarzania lub bez "godziwego" zysku. Wysoka Komisja uznała, że japońskie firmy zaniżają ceny o:

Alps Electrical Co. Ltd.	- 7,4%
Brother Industries Ltd.	- 45,8%
Citizen Watch Co. Ltd.	- 55,0%
Copal Co. Ltd.	- 18,6%
Fujitsu Ltd.	- 86,0%
Japan Business Computer Co. Ltd.	- 22,4%
Nakajima Ltd.	- 12,3%
NEC Corporation	- 70,0%
OKI Electric Industry Co. Ltd.	- 9,2%
Seiko Epson Corporation	- 33,6%
Seikosha Co. Ltd.	- 73,0%
Shinwa Digital Industry Co. Ltd.	- 10,5%
Star Micronics Co. Ltd.	- 13,6%
Tokyo Electric Co. Ltd.	- 4,8%
Tokyo Juki Industrial Co. Ltd.	- 83,5%

Początkowo zamierzano produkty wymienionych firm obciążyć cłem w podanej wysokości (dotąd wynosiło ono 4,9%). Po rozmowach handlowych, a zapewne i dyplomatycznych, z partnerem tak silnym jak Japonia obrońcy Wspólnego Rynku spuścili nieco z tonu i ustalili cło w wysokości 33,4%.

Firmy, które na "czarnej liście" miały mniej niż owe 33,4% dumpingu, obciążone zostały cłem w wysokości podanej w tabeice. Dla przykładu wytwórca najpopularniejszych na naszym rynku drukarek - firma Star płaci obecnie 13,6% cła, udowodniła bowiem, że koszty produkcji, po uwzględnieniu godziwego zysku, ma naprawdę niskie.

W efekcie wpływ nowych ceł na ceny jest umiarkowany: w RFN produkty NEC i EPSON podrożały o ok. 15%, STAR o ok. 13%. We Włoszech i Francji ceny wzrosły o ok. 5% (ale tam było drogo już dawniej).

Omawiana zapora celna wprowadzona została od czerwca i ma obowiązywać na razie przez cztery miesiące. W tym czasie firmy mogą odwoływać się, z możliwości tej skorzystały już NEC i EPSON. Japońscy producenci nie ujęci w tym wykazie płacą, jak poprzednio, 4,9%.

Powiedzonko jest chińskie, ale posunięcia EWG przypominają "papierowego tygrysa", zwłaszcza na tle tempa ogólnego wzrostu cen na naszym rynku. Nawiasem mówiąc nowe cła nie wpłyną na ceny drukarek przechodzących przez obszar EWG tranzytem, a więc sprzedawanych przez firmy eksportowo-wysyłkowe zaopatrujące się prosto od producentów (np. ABC Data).

M.A.M.

IBM PS/2 model 70

Błękitny gigant podjął próbę powrotu na czoło stawki w klasie 386, a w ostatnich miesiącach sprzedaż komputerów firmy Compaq (model Deskpro 386) znacznie przewyższała sprzedaż modelu 80 rodziny PS/2.

Najnowszy model PS/2, mimo numeru pośredniego między 60 a 80 jest w tej chwili niewątpliwie flagowym okrętem koncernu. W obudowie modelu 50 (klasycznej, leżącej na biurku) mamy płytę główną z procesorem 80386 pracującym z zegarem 25 MHz, koprocesorem (80387 pracującym również z zegarem 25 MHz. O niezwykłym tempie pracy nowego modelu decyduje na równi z tempem zegara zupełnie nowy sterownik napędów dyskowych, który na przykład dyskietkę (1,44 MB, 3,5 cala) czyta z tzw. przeplotem 1:1, co znacząco przyspiesza operacje dyskowe.

Inną nowością jest układ zarządzający dostępem do magistrali 80325, który umożliwia pracę mikroprocesora bez stanów oczekiwania, gdyż magistrala pracuje niezależnie od procesora.

Rodzina PS/2 zaczyna równocześnie wzbogacać się o kuzynów. Firma Chips Technology opracowała zestaw kości integrujący procesor i zarządzanie magistralą Microchannel. Korzystają z niego firmy Dell (model 500) i Tandy w swym najnowszych produktach.

Za miesiąc w Komputerze:

Test: - Drukarka Star LC-10 Colour

Sprzęt: - Nowe mikrokomputery firmy Schneider
- Najmniejszy PC: Husky Hawk 8/16

Programy: - Pogoń za mikrosekundami w Turbo-Pascalu
- Polskie znaki z kalwiatury IBM PC
- SynGraph: program graficzny dla Atari XE/XL
- "Speedscript" z polskimi literami
- TEX dla autorów tekstów naukowych
- Flash + dla ZX Spectrum.
- Opisu CP/M część trzecia
- Rezydujące rozszerzenia systemu CP/M
- Praktyka programowania: uruchamianie programu

Stragan: - Polska klawiatura dla Atari ST
- Atari ABAQ

Wywiad: - z Mariuszem Olechem z Hamburga

"Na 10 dni przed drukiem" przygotował 12.07.88 Władysław Majewski, korzystając z programu Signum II, Atari ST i drukarki Star NB 24-15, pozostającej pod opieką autoryzowanego serwisu drukarek STAR, firmy **COPACT** Sp. z oo., Marki, Świerczewskiego 15

\Kurier\

Zbigniew Blewoński

AutoCAD w Polsce

Powoli wyrastamy z zakaźnej choroby wieku dziecięcego komputeryzacji - powszechnego piractwa programowego. Ci, którzy chcą profesjonalną działalność biura projektów oprzeć np. na pakiecie typu AutoCAD, coraz częściej rozumieją, że otrzymywana po zakupieniu legalnej kopii programu pełna dokumentacja oraz pomoc autorów mają bardzo duże znaczenie.

Mówi o tym Richard Handyside, wiceprezydent europejskiego oddziału firmy Autodesk, dyrektor zarządzający jej brytyjskiego biura.

Po raz pierwszy nasza redakcja ma okazję poznać przedstawiciela firmy Autodesk, prosimy więc o kilka słów na temat historii firmy.

Firma powstała w Kalifornii w kwietniu 1982 roku. Założycielami byli w większości wysokiej klasy programiści - 11 osób z USA i 4 z Europy *(wśród nich nasz rozmówca - red.).* Dziś w filiach na całym świecie zatrudniamy 400 osób. Biura znajdują się w USA, Anglii, Szwajcarii, Szwecji, Japonii i Australii. Produkujemy oprogramowanie powszechnego użytku.

Program AutoCAD zaprezentowany został po raz pierwszy na wystawie komputerowej w końcu 1982 roku. Na rynek wszedł na początku 1983 roku i od tego momentu sprzedano około 150 000 jego kopii, co stanowi ponad 50% sprzedanego na świecie oprogramowania typu CAD. Podkreślam, że podane liczby dotyczą kopii sprzedanych legalnie. Od samego początku AutoCAD był głównym, choć nie jedynym produktem naszej firmy.

Z czasem AutoCAD stał się światowym standardem. Jakie są źródła tego sukcesu?

Program powstał jako tanie narzędzie wykorzystujące możliwości IBM PC do komputerowo wspomaganego projektowania (CAD). Wówczas tzw. tanie systemy CAD (dla dużych komputerów) kosztowały od 50 000 do ponad 250 000 funtów. Stanowisko pracy dla AutoCAD-a natomiast (wraz z małym ploterem) nieco ponad 6000 funtów.

AutoCAD tworzony był od początku jako system otwarty - umożliwiający użytkownikowi przystosowanie go do swoich potrzeb i przyzwyczajeń. Można łatwo zmieniać menu, budować biblioteki własnych krojów pisma, symboli, rodzajów linii itp., a korzystając z języka AutoLISP tworzyć tzw. nakładki do specjalistycznych zastosowań. Program dostępny jest w kilku wersjach językowych - niemieckiej, francuskiej, hiszpańskiej, włoskiej i japońskiej. Przygotowywana jest wersja duńska i - w zależności od zapotrzebowania - rosyjska oraz polska.

Czyżby perspektywy sprzedaży AutoCAD-a na rynku Europy Wschodniej, a zwłaszcza polskim, zapowiadały się aż tak pomyślnie?

Już od 18 miesięcy interesujemy się bliżej rynkiem Europy Wschodniej jeżdżąc na organizowane tu wystawy. W ubiegłym roku uczestniczyliśmy w kilkunastu. Na każdej obserwowaliśmy duże zainteresowanie naszym programem. Co więcej, z rozmów z polskimi użytkownikami pirackich kopii wynika, że chcieliby kupić legalne kopie, pełną dokumentację oraz otrzymywać informacje o oprogramowaniu dodatkowym, nowych wersjach itp. Jesteśmy zainteresowani powstaniem w Europie Wschodniej normalnego rynku oprogramowania i naszą na nim obecnością. Jeśli jednak utrzyma się obecna sytuacja, w której większość oprogramowania to pirackie kopie - nie będziemy tracić czasu.

Od czego Autodesk zaczyna na nowym rynku?

Etap początkowy to rozmowy mające na celu stworzenie autoryzowanych przez nas ośrodków szkolenia użytkowników AutoCAD-a. Pod koniec ubiegłego roku np. powstał taki ośrodek w Moskwie, a w ubiegłym tygodniu przeszkoliliśmy 14 "nauczycieli AutoCAD-a", którzy umożliwią utworzenie 6 centrów szkoleniowych w Czechosłowacji. W Polsce ośrodek taki tworzy firma APLIKOM z Łodzi.

W jaki sposób zorganizowane jest szkolenie?

Szkoleniem użytkowników programu zajmują się uczelnie, szkoły oraz różne małe firmy - w Anglii jest ponad 350 takich ośrodków.



Natomiast Autodesk prowadzi kursy tylko dla organizatorów i nauczycieli wspomnianych ośrodków oraz kursy dla sprzedawców-pośredników (ang. *dealer* - *red.*). Warto też wspomnieć, że na wielu wydziałach inżynierskich i architektonicznych szkół wyższych w Wielkiej Brytanii nauka posługiwania się AutoCAD-em jest planowym elementem toku studiów. Wszak większość absolwentów będzie w przyszłości korzystała w codziennej pracy z systemów CAD.

Często wizytujemy organizacje prowadzące szkolenia, kontrolujemy poziom wykładowców, pozostawiając jednak wolną rękę co do metod. Organizowane są różne kursy - dla początkujących, dla zaawansowanych, dla zastosowań specjalistycznych itp. Z naszych doświadczeń wynika, że najlepsze rezultaty osiągane są po 2-3 -dniowej intensywnej sesji wykładowej, kilkutygodniowej lub kilkumiesięcznej przerwie na własną pracę z programem oraz ponownej sesji w ośrodku. W Anglii zabiegamy o wprowadzenie "egzaminu państwowego", który byłby ogólnie uznawanym świadectwem umiejętności korzystania z AutoCAD-a. Chcemy, by szkolenia w Polsce prowadzone były według podobnego schematu.

Ile kosztuje AutoCAD i czy firma będzie go sprzedawać w Polsce za złotówki?

Jedna licencjonowana kopia programu (zabezpieczona kluczem sprzętowym) kosztuje w Anglii 2 500 funtów. Dla uczelni, szkół i innych organizacji wykorzystujących AutoCAD do nauczania, a nie do produkcji, stosowana jest 80% zniżka. Przy zakupie większej liczby kopii również udzielamy rabatu, ale nie tak dużego, jak można

> 6

Polski przedstawiciel firmy Autodesk

Autoryzowanym przedstawicielem firmy Autodesk w Polsce sprzedającym za złotówki jest Przedsiębiorstwo Produkcyjno-Handlowe Zastosowań Informatyki APLIKOM spółka z o.o.

Od początku marca spółka organizuje regularne cykle szkoleniowe dla użytkowników początkujących i zaawansowanych (początkowo tylko audytoryjne). Uruchomione zostaną także konsultacje dla zastosowań specjalnych. Firma rozpowszechnia autoryzowane kopie AutoCAD-a. Pierwszą proponowaną ceną jest 9 mln złotych, ale prowadzone są rozmowy z firmą Autodesk zmierzające do jej obniżenia oraz zmian zasad sprzedaży - z autoryzacji poszczególnych kopii na autoryzację użytkownika, co pozwala na eksploatację programu na kilku stanowiskach. Będzie także utrzymana 80% zniżka przy zakupie programu przez uczelnie i inne organizacje wykorzystujące go tylko do celów szkoleniowych. Wspólnie z firmą APLIKOM Autodesk przygotuje polską wersję programu (z uwzględnieniem polskich norm dotyczących wymiarowania rysunków technicznych) i oryginalną polską dokumentację.

by się spodziewać - powodem jest stosunkowo niska cena pierwszej kopii.

Autodesk nie prowadzi bezpośrednio sprzedaży detalicznej. Oprogramowanie trafia do użytkowników poprzez autoryzowanych dystrybutorów, w praktyce nie udzielamy także wyłączności do reprezentowania naszych interesów na danym obszarze. W Polsce takim dystrybutorem jest firma RANK XEROX - sprzedająca za waluty wymienialne oraz wspomniana już firma APLIKOM - sprzedająca za złotówki i decydująca o cenie.

Jaki jest stosunek firmy do powstających nakładek na AutoCAD-a? Czy są one w jakikolwiek sposób weryfikowane?

AutoCAD jest tylko programem rysującym, czyli narzędziem bardzo uniwersalnym. Inżynier w swojej codziennej pracy projektowej potrzebuje często bardzo specjalistycznego narzędzia do ściśle określonych prac - rolę tę bardzo dobrze spełniają tzw. nakładki, czyli specjalne menu i programy w języku AutoLISP. Producentem i dystrybutorem takiego programu może być każdy posiadacz licencjonowanej kopii AutoCAD-a. Ze swej strony zachęcamy do pisania tych nakładek. Regularnie publikujemy listę znanych nam programów (obecnie ponad 500). Nakładek tych w żaden sposób nie kontrolujemy, całą odpowiedzialność za ich działanie ponoszą autorzy. Myślę, że nie ma żadnych przeszkód, by powstające w Polsce programy tego typu znalazły się na naszej liście i były sprzedawane na światowym rynku.

Czy oferowana obecnie wersja 9. zawiera jakieś istotne zmiany (co sugerowałby skok w numeracji)?

Wersja 9. sprzedawana jest od listopada 1987 i jest następcą wersji 2.6. Zachowana została całkowita zgodność danych z poprzednimi wersjami. Obecna zapewnia także bezpośrednią wymienialność danych (plików) pomiędzy programami (AutoCAD) działającymi w różnych systemach - MS - DOS, Unix, Xenix, VMS. Notabene we wszystkich tych systemach AutoCAD jest z punktu widzenia użytkownika identyczny. W wersji tej poprawiono także komunikację użytkownik-program poprzez wprowadzenie dodatkowego *"pull down menu"* i *"icon menu"*, wygodnych szczególnie dla początkujących użytkowników. Tak więc zależnie od stopnia zaawansowania i przyzwyczajeń użytkownik może sterować programem z tego właśnie menu, z menu ekranowego znanego z poprzednich wersji, z menu *digitizera* lub pisząc komendy bezpośrednio z klawiatury. W wersji tej dostępne jest nowe narzędzie (tzw. *Autodesk Device Interface - ADI*) pozwalające użytkownikom i producentom sprzętu na łatwe pisanie programów obsługi nietypowych urządzeń graficznych.

Jakie są przewidywane kierunki dalszego rozwoju programu?

Jeszcze przed końcem tego roku ukaże się trójwymiarowa wersja uwzględniająca zasady perspektywy. Bardzo intensywnie pracujemy nad programami dla komputerów SUN Microsystem 2 i 3 w systemie Unix. Obecna wersja programu wyczerpała już możliwości systemu PC/MS-DOS i możliwości sprzętowe tradycyjnych komputerów IBM PC XT/AT: nie można już przyspieszyć pracy programu (szczególnie przy edycji dużych rysunków), mogą występować kłopoty z dużymi programami w AutoLISP-ie, utrudniona jest współpraca z programami rezydującymi w pamięci np. SideKick. Z wymienionych powodów nie należy spodziewać się znacznych zmian w tej (dla MS-DOS) wersji programu. Przyszłość widzimy w systemach Unix i OS/2.

Dziękujemy za rozmowę i życzymy powodzenia na polskim rynku.

AutoCAD wersja 9
- ulepszona komunikacja z użytkownikiem - całkowicie definiowane menu (obrazkowe, rozwijane, ekranowe i menu dla digitizera);
- możliwość redefinicji wszystkich komend programu;
- Autodesk Device Interface (ADI) - narzędzie do budowy tzw. driverów dla nietypowych urządzeń graficznych współpracujących z AutoCAD-em;
- pliki rysunkowe mogą być bezpośrednio przenoszone między wersjami AutoCAD-a pracującymi w systemach PC/MS-DOS, Unix, VMS;
- 19 standardowych krojów pisma m.in. greka, cyrylica, gotyk i pismo odręczne (kaligraficzne);
- zwiększone możliwości języka AutoLISP.

\Kurier\

Czytaj!

Stanisław Borak, Jerzy Klaczak, Stanisław Korczak, Zdzisław Płoski "System operacyjny George 3", WNT 1987, wyd. II poprawione, 5800 + 200 egz., 338 str., 370 zł, seria "Biblioteka Inżynierii Oprogramowania".

Tę pozycję jedynie zasygnalizuję, nie wydaje się bowiem, aby krąg jej odbiorców był zbyt szeroki. George 3 to system operacyjny stosowany w maszynach cyfrowych ODRA 1305. Książka jest podręcznikiem przeznaczonym dla użytkowników maszyn produkowanych przez Elwro, dlatego też opisuje standardowy system operacyjny w wersji dostarczanej przez Elwro nabywcom instalacji komputerowych.

Wertując książkę nie od razu zwróciłem uwagę na czas i miejsce podane przez Autorów w przedmowie. Kiedy już przeczytałem to zdanie, nie mogłem uwierzyć. Brzmiało ono: "Wrocław, w lutym 1979 oraz w lipcu 1984". Sprawdziłem z grubsza, zgadzało się: oddano do składania 24.01.1985, druk ukończono w październiku 1987. Komentarz zbyteczny. Może warto tylko uświadomić sobie, jaką drogę przeszła światowa informatyka od lipca 1984 do lutego 1988 (otrzymałem książkę do zrecenzowania).

Rolf Hedtke "Systemy mikroprocesorowe. Niezawodność, testowanie, tolerancja błędów", tłumaczył Andrzej Dworak, WNT 1987, wyd. I, 6800 + 200 egz., 175 str., 500 zł, seria "Układy i systemy elektroniczne".

Wydana w 1984 roku w Berlinie i Heidelbergu książka z pewnością warta jest odnotowania, a projektanci i użytkownicy systemów mikroprocesorowych oraz studenci informatyki i elektroniki znajdą w niej cenne wskazówki, jak należy projektować niezawodne systemy informatyczne. Wielu projektantów stwierdzało nieraz z rozczarowaniem, że systemy mikroprocesorowe ulegają uszkodzeniom częściej niż oczekiwano. Co gorsze, w odróżnieniu od układów zbudowanych z elementów dyskretnych, błędy spowodowane niewłaściwą konstrukcją powodują najczęściej całkowitą awarię mikrokomputera. Nie jest również możliwe bezbłędne opracowanie większych programów, najczęściej nie można nawet kompletnie przetestować programu. Nic więc dziwnego, że - jak podaje autor - w nowo projektowanych systemach 50% awarii powodują błędy oprogramowania możliwe do zauważenia dopiero w czasie stosowania systemu.

Obydwu zagadnieniom poświęca autor obszerne analizy teoretyczne, by dojść w konkluzji do praktycznych technik i sposobów zapobiegania i korygowania błędów. Będą to środki dla ochrony przed skutkami uszkodzeń podzespołów, metody testowania układów scalonych, wpływ projektowania sprzętu i oprogramowania na niezawodność systemu, stosowanie technik redundancyjnych, kody wykrywania i korygowania błędów.

S.M.K.

\Kurier\

Stanisław Marek Królak

Terminator terminologiczny [13]

Miesiąc temu zajmowaliśmy się zagadnieniami ogólnymi, dzisiaj pora na rozwinięcie szczegółowe i rachunek sumienia. Nie da się bowiem ukryć, że grzechy terminologiczne wdzierają się również na łamy "Komputera". Zwykle bywa tak, że autorzy piszący do nas posługują się okropnym żargonem, w którym mieszanina słów angielskich i spolszczonych terminów informatycznych tworzy bełkot zrozumiały tylko dla wtajemniczonych. Kierownicy działów skupiają swoją uwagę na adiustacji merytorycznej nie przejmując się nadmiernie językową, bo przecież jest jeszcze sekretarz. Tyle tylko, że poprawiając olbrzymią ilość błędów nie jestem w stanie nie pominąć żadnego (errare humanum est). I tu zaczyna się już nieszczęście, bo przepuszczone przeze mnie błędy ukazują się na łamach.

Przypomniał mi o tym pan Jerzy Battek z Wrocławia, jeden z tych - jak pisze - co to na informatyce dojadają resztki zębów. Dziękuję za sympatyczny list i elegancką uwagę, iż to co w nim krytyczne, napisane zostało po to, aby pomóc redakcji i nie musi być drukowane. Uwagi naszego Czytelnika są jednak na tyle ważne, że fragmenty listu zamieszczam, a myślę, że Szanowny Korespondent mi to wybaczy.

"Nie jestem purystą językowym, ani chorym na terminologię, ale do języka informatycznego wdarło się trochę niechlujstwa, przede wszystkim za sprawą niedouczonych tłumaczy różnych instrukcji z języka angielskiego, którzy to "tłumacze" nie znali nie tylko języka angielskiego, ale co gorsze także języka polskiego. Kilka przykładów najbardziej rażących.

1. Zaczyna się panoszyć (również w "Komputerze") nazywanie wierszy programu (który jest wszak tekstem składającym się z wierszy) liniami. Słowo linia kojarzy się w języku polskim z geometrią, geografią, techniką, polityką, gospodarką itd. Z tego, że wiersz tekstu nazywa się po angielsku *line*, nie wynika, że ma się po polsku nazywać linią. Nawet termin *steering lines* jakiś matoł przetłumaczył na linie sterujące, co brzmi już całkiem absurdalnie i kojarzy się z jakimiś obwodami elektrycznymi, a nie napisami lub zdaniami. Zachowajmy zatem wiersze programu, a uparci niech mówią przynajmniej linijki, a nie linie.

2. Ktoś (też matoł) przetłumaczył termin *double precision* na podwójną precyzję, bo nigdy nie słyszał, że w matematyce numerycznej używa się od zawsze terminu **dokładność**, w tym przypadku **podwójna dokładność**.

3. Termin *linker* i *linkować* używany przez Amerykanów ma polski odpowiednik (wywodzący się zresztą też z angielskiego) **konsolidator** i **konsolidować**. Słowo trochę dłuższe, ale zawierające jakąś treść. (...)

Myślę, że "Komputer" ma dostateczny autorytet, by zmobilizować czytelników i korespondentów do używania poprawnej terminologii, choćby poprzez dawanie przykładu. Wszak każdy szanujący się Polak musi przeciwstawiać się żargonowi typu *na ile ten winogron* lub *przyjmuję się do pracy w*, co ma oznaczać *podejmuję pracę w*."

No cóż, mam nadzieję, że autorytetu wystarczy i w komputerowym światku zamiast ***lines*** będą ***wiersze***, zamiast ***precision*** - ***dokładność*** i że nie będzie się już nic linkować przy pomocy linkera, lecz **konsolidować** program za pomocą **programu konsolidującego**. Ponieważ jednak przed nami daleka droga, nie raz wrócimy do tematu.

Pewnie już za miesiąc.

\Kurier\

Komputeryzujemy się

Zaintrygowani tytułem "Komputerowy rasizm?" przeczytaliśmy artykuł w "Trybunie Opolskiej" - i czego to się człowiek nie dowie o sobie... Ten "komputerowy rasizm" mianowicie uprawia jakoby nasz miesięcznik!

Otóż, zdaniem uczestników zjazdu użytkowników Commodore, który się odbył w Opolu, "*adoratorzy Commodore są dyskryminowani w najbardziej liczącym się na rynku wydawniczym tej branży czasopiśmie "Komputer". Pismo to lansuje, mimo protestów czytelników, firmę Atari i jej wyroby.*"

Wyjaśnialiśmy wielokrotnie już, że poszczególnymi typami komputerów staramy się zajmować proporcjonalnie do ich rozpowszechnienia w Polsce - w innym przypadku stracilibyśmy przecież czytelników. Z rasizmem ma to tyle wspólnego co z plamami na słońcu, czyli nic. Zabawny atak "Trybuny Opolskiej" dowodzi jednak jak bardzo jeszcze młodziutka jest ta nasza "cywilizacja komputerowa", skoro rodzą się tak dziecinne reakcje. Dojrzałość przyjdzie z wiekiem, zresztą pewien postęp już istnieje, bowiem - jak pisze dalej ta sama gazeta - "*tym razem, odwrotnie niż w roku ubiegłym, nie programowano komputerów tak, by obrzucały na ekranach monitorów błotem konkurencję spod znaku Atari.*"

* * *

Przedstawicielka "Kobiety i Życia" zwiedziła giełdę komputerową. "*Warszawska giełda, działająca w szkole przy ul. Grzybowskiej, jest zdominowana przez nastoletnich chłopców (...) Na giełdzie istnieje spora konkurencja i trwa walka o konsumenta. Toteż widać pewną staranność w reklamowaniu towaru. Chłopcy robią katalogi, mają plansze, zabiegają o estetyczny wygląd swego stoiska i eksponowane miejsce ... Gry komputerowe na giełdzie są tańsze od gier sprzedawanych w Składnicy Harcerskiej i w Peweksie. Żeby więc zarobić, trzeba mieć wysokie obroty.*

Na giełdzie można się też uczyć. Chłopcy wymieniają informacje, dyskutują. Nie znaczy to jednak, że wystarczy chodzić na giełdę, by zostać wybitnym informatykiem. Raczej można się tam wykierować na zręcznego handlowca."

* * *

"*Według ostrożnych szacunków, na naszym rynku wewnętrznym operuje ok. 250 firm, głównie prywatnych, które proponują właściwie wszystko: supersprzęt, oczywiście kompatybilny z IBM, programy, szkolenie, doradztwo, naprawy gwarancyjne, części, skup itd.* - podaje "Rzeczpospolita". - *Rynek ten obraca ogromnymi sumami - samego sprzętu elektronicznego przybywa nam rocznie za mniej więcej 100 mld zł.*"

"*Mamy w Polsce* - powiada na łamach dziennika prof. Michał Kleiber z Instytutu Podstawowych Problemów Techniki PAN - *bardzo dużo mikrokomputerów, jeśli porównać chociażby z innymi krajami socjalistycznymi. Trzeba je jednak profesjonalnie wykorzystać. Właśnie pisanie profesjonalnego oprogramowania oraz konfigurowanie sprzętu dla konkretnych zastosowań to jedyna droga, aby nadać całej tej informatyzacji sens.*"

* * *

Nasz Zwykły, Prosty Przedsiębiorca - Importer, odbywający swe mozolne wyprawy do Singapuru i z powrotem, został dostrzeżony nawet na drugiej półkuli, czemu daje wyraz pismo "Wall Street Journal", a co przytaczamy za "Trybuną Robotniczą":

"*Kapitalistyczne komputery stały się wysoko atrakcyjnym towarem dla zręcznych ludzi interesu, dążących do osiągnięcia pokaźnych zysków w skomplikowanym świecie zakazów handlowych, licencji eksportowych i czarnorynkowych kursów walut. Większość Polaków, aby zarobić tyle, ile zarabia na jednej transakcji indywidualny importer, potrzebuje 10 lat.*"

Gazeta katowicka cytuje te słowa bez szczególnej aprobaty dla "zręcznych ludzi interesu". W ustach "Wall Street Journal" - jest to sto lat prawie liczący dziennik amerykańskiego wielkiego businessu - są one jednak niewątpliwie dowodem prawdziwego uznania. Robić tak dobre interesy i to w TAK SKOMPLIKOWANYM ŚWIECIE - doprawdy, że po tych Polakach można spodziewać się wszystkiego...

* * *

Można się spodziewać, niestety, także tego, że skopiują cudzy program i nie zapłacą. "*Ci, którzy zdecydowali się wydać dewizy, aby kupić oprogramowanie u producenta, nie mogą tego uczynić, gdyż ów producent odmawia sprzedaży swych programów do Polski. I nie łudźmy się, że w grę wchodzą ograniczenia embargowe. Znam wiele przypadków, gdy w odpowiedzi na ofertę zakupu producent wyraźnie stwierdził, że nie sprzeda swoich programów do kraju, w którym nie podlegają one żadnej ochronie*" - mówi prezes Polskiego Towarzystwa Informatycznego prof. Andrzej Blikle w rozmowie z red. Magdą Sowińską w "Polityce".

"*Na rynku mikrokomputerowym (...) panuje w przeważającej mierze oprogramowanie kradzione (...) Dzieje się tak, ponieważ brak jest w Polsce wyraźnych przepisów chroniących prawa wytwórców oprogramowania. Sytuacja ta powoduje nie tylko konflikt sumienia u wielu informatyków, ale również bardzo poważne szkody ekonomiczne, uniemożliwiając powstanie polskiego przemysłu oprogramowania.*

(...) Znajdujemy się w błędnym kole: ponieważ wszyscy kradną, nikt nie zaryzykuje produkcji, a ponieważ nikt nie produkuje, na rynku dostępny jest jedynie towar kradziony. Nie ma innego wyjścia jak natychmiastowe uregulowanie strony prawnej..."

J.R.

\Kurier\

Postaci mikroświata SEYMOUR PAPERT

Profesor matematyki i pedagogiki MIT, twórca Logo, członek honorowy licznych stowarzyszeń oraz uczestnik wielu międzynarodowych kongresów przyjechał do Bostonu w 1964 r. po pięciu latach intensywnej pracy nad naturą myślenia u dzieci. Przeprowadzka z alpejskiej, podgenewskiej miejscowości do zurbanizowanego i skomputeryzowanego świata wpłynęła na jego dalsze zainteresowania naukowe, które zaowocowały m.in. rozwinięciem modelu przetwarzania informacji wywodzącego się ze



strategii myślenia. Za ważne i perspektywiczne z punktu widzenia rozwoju nauki o myśleniu uznał Papert dostarczenie dzieciom wszystkiego, co najlepsze z wiedzy o komputerach, włączając w to i technologię, i idee. To on jest autorem hasła: "komputer jak ołówek". Rozumiał przez nie możliwości, jakie komputer powinien w przyszłości stwarzać dzieciom, a więc zarówno ułatwiać gryzmolenie, jak i kaligrafowanie, bazgranie i kreślenie, notowanie i prowadzenie systematycznych opisów. W praktyce chodziło o udostępnienie dzieciom sprzętu oferującego lepszą grafikę i łatwy kontakt z użytkownikiem. W dążeniach, by w edukacji nie opierać się na prymitywnych komputerach, poparł Paperta jedynie zespół Alana Kaya (Xerox Palo Alto Research Center). Niestety żaden terminal w latach siedemdziesiątych, ani późniejszy domowy TRS-80, wizji "komputera jak ołówka" przybliżyć nie mogły.

O języku komputerowym dostępnym dla dzieci zaczął myśleć Papert w roku 1967. Nie chodziło mu o język-zabawkę, lecz o profesjonal-

> 8

ny język programowania zrozumiały dla użytkownika o zerowym przygotowaniu matematycznym. Ideę tę szybko podchwyciła Educational Technology Group (Bolt Beranek i Newman), tworząc pierwszą implementację języka Logo. W roku szkolnym 1968/69 dwunastu siódmoklasistów z Lexington korzystało już z niej na lekcjach matematyki.

Pierwsza wersja Logo nie miała grafiki, uczniowie pisali programy tłumaczące angielski tekst na szkolny żargon, programy generujące rymy oraz gry. Dla Paperta było to potwierdzeniem faktu, iż językiem Logo mogą posługiwać się zupełni nowicjusze. Chcąc zainteresować nim dzieci młodsze, zaproponował wyposażenie go w programowanego żółwia.

Twórcy Logo udało się niebawem zorganizować przy Laboratorium Sztucznej Inteligencji oraz Laboratorium Nauk Komputerowych w Massachusetts Institute of Technology - Laboratorium Dziecięce. Tu właśnie, obserwując reakcje uczniów, dokonano szeregu poprawek i udoskonaleń w kolejnych wersjach Logo. Główne założenia języka oraz efekty jego stosowania w pracy z dziećmi omawia Papert w swej najbardziej znanej książce - "Mindstorms: Children, Computers and Powerful Ideas" (1980).

Dziś Papert kieruje dwoma programami badawczymi. Pierwszy, realizowany od 1985 r. w Boston's Project Headlight, służyć ma rozwojowi kultury komputerowej uczniów klas III-V. W eksperymencie prowadzonym w Henningan Elementary School 220 uczniów dysponuje 100 komputerami, przy których może spędzać do 2 godzin dziennie. Papert uważa bowiem, że o komputerowej rewolucji w nauczaniu będzie można mówić dopiero wówczas, gdy komputery staną się tak dostępne jak ołówki. "Obecnie w amerykańskich szkołach około miliona komputerów przypada na 50 milionów uczniów. Co można robić, dysponując 1/50 częścią komputera? Nawet w Bostonie w kolejce do klawiatury stoi 18 uczniów. Czy potraficie sobie wyobrazić posługiwanie się jednym ołówkiem przez 18 osób?"

(dm)



Wacław Z. - pionier mikrokomputeryzacji

Kurier

Nowości

Komputerowe diagnozowanie chorych

Ministerstwo Zdrowia ZSRR posiada "Zautomatyzowany system badań profilaktycznych ludności" (ASPON), umożliwiający w ciągu jednej zmiany poddanie rutynowym badaniom 120 osób. Komputer zbiera dane o pacjentach, przetwarza je, po czym opracowuje diagnozę. Informacje te, gromadzone przez dłuższy czas, pomocne są także w wykrywaniu tzw. czynników ryzyka, charakterystycznych dla danego rejonu czy miejscowości.

ASPON jest częścią większej bazy danych medycznych, obejmującej kilkaset miejscowości. Składa się ona z kilku podsystemów. "Rejestracja" i "Antropometria" zbierają informacje o bieżącym stanie organizmu oraz o przebytych chorobach, "Krwiobieg", "Układ oddechowy", "Wzrok", "Słuch" gromadzą dane o konkretnych narządach człowieka. Na przykład "Krwiobieg" rejestruje i opracowuje elektrokardiogram, mierzy i zapisuje puls oraz ciśnienie tętnicze. Wyniki badań laboratoryjnych wprowadzane są do podsystemu "Laboratorium". System po zebraniu danych przetwarza je i na ich podstawie dokonuje podziału pacjentów, wyodrębniając zdrowych, tzw. grupę ryzyka (z pewnymi nieprawidłościami w działaniu jakichś narządów, ale nie prowadzącymi jeszcze do choroby) oraz chorych, wśród których wyróżnia 16 podgrup w zależności od rodzaju dolegliwości. Do osobnej grupy trafiają chorzy wymagający opieki szpitalnej.

Działający od 1983 r. na Litwie w ramach ASPON "Kompleksowy zautomatyzowany system badań medycznych ludności" (KASMON) realizowany jest na mikrokomputerach Iskra-1256 i Iskra-226. System obsługuje jeden lekarz oraz 9 osób pomocniczego personelu medycznego. W 1984 r. KASMON objął badaniami 92,2 % ludności republiki. U ponad 12 % badanych wykryto symptomy chorób, których istnienia pacjenci nie odczuwali. Najbardziej przydatna okazała się komputerowa analiza badań elektrokardiologicznych. Stworzenie w pamięci komputera swoistego archiwum elektrokardiogramów dużej liczby pacjentów ułatwiło szybką diagnozę różnorodnych chorób serca.

Jak informuje "Informatika i obrazowanije" - ze względu na nowość tego typu badań w Związku Radzieckim - diagnozy stawiane przez KASMON dublowane są przez lekarzy metodami tradycyjnymi. Zgadzają się w ponad 90 % przypadków.

(dm)

AIDS i komputery

Znaczenie wszechstronnej informacji jest w walce z AIDS oczywiste. Russell Toth, pracownik Centrum Pomocy Homoseksualistom w Los Angeles, zorganizował przy finansowym wsparciu kalifornijskiego departamentu zdrowia komputerową sieć informacyjną o AIDS (Computerized AIDS Information Network -CAIN). Jest ona częścią komercjalnego systemu informacyjnego Delphi, prezentującego dane encyklopedyczne, aktualności, prognozę pogody, wiadomości sportowe i towarzyskie. Stąd też podłączenie modemu do systemu CAIN jest płatne (50$), koszt korzystania z niego wynosi od 11 do 16 centów za minutę. Miesięcznie przybywa ponad 500 nowych użytkowników (w roku 1984 było 2). CAIN oferuje im także (za cenę przesyłki) broszury, poradniki, prenumeratę "AIDS Treatment News" (w kopertach bez nadruku). System wykorzystuje informacje z prasy medycznej, Associated Press, Amerykańskiego Centrum Statystyki Medycznej (m.in. opisanych 41 825 przypadków choroby).

Dla osób nie dysponujących odpowiednim sprzętem komputerowym CAIN udostępnił także informację telefoniczną.

(dm)

ASEMBLER SKROŚNY ZNAD RENU

Christian Franke, programista z Akwizgranu, opracował makroasembler MAS68X przeznaczony dla komputerów kompatybilnych z IBM PC. Za jego pomocą można przygotować relokowalny kod dla wszystkich mikroprocesorów z rodziny 68000. (adan)

80386 W KOMPUTERACH PRZENOŚNYCH

Jednym z pierwszych komputerów przenośnych wykorzystujących mikroprocesor 80386 jest DRV LCD-386. Waży 10,5 kg. Pamięć masową stanowi 3,5-calowy dysk twardy o pojemności 64 MB. Maszyna wyposażona jest w stację dysków 5,25" standardu AT i ekran ciekłokrystaliczny.

Niewiele lżejszy (9 kg) jest Compaq 386-Laptop wyposażony w ekran plazmowy o rozdzielczości 640x400 punktów, który jednak zużywa tyle energii, że maszyna nie może być zasilana z baterii. Pamięć operacyjna - 1 MB. Pamięć masowa na dyskach 5,25" oraz winchester 40 lub 100 MB.

Toshiba proponuje komputer podręczny T5100 stanowiący kontynuację linii rozwojowej T3000. Pamięć operacyjna 2 MB może być wewnętrznie rozszerzona do 4 MB. Stacja dysków 3,5", dysk twardy 40 MB i ekran plazmowy o rozdzielczości 640x400 punktów. Wszystkie opisane maszyny kosztują w RFN od kilkunastu do ponad 20 tys. marek.

(adan)

Kurier

Andrzej Załuski

Salmed '88

Już po raz siódmy na terenach Międzynarodowych Targów Poznańskich w kwietniu b.r. odbył się Międzynarodowy Salon Medyczny "Salmed'88". W targach brało udział około 350 firm z 20 państw. W stosunku do targów poprzednich zdecydowanie wzrosła liczba krajowych ofert obejmujących zastosowania komputerów w medycynie. Na nich skupimy naszą uwagę. Prezentowane rozwiązania podzielić można na 4 grupy tematyczne:

1. Zestawy do analizy obrazowej,

2. Zestawy do analizy przebiegów EKG i EEG,

3. Zestawy do archiwizacji i przetwarzania danych oraz systemy ekspertowe,

4. Inne rozwiązania z zastosowaniem mikroprocesorów lub mikrokomputerów.

* * *

Komputerowa analiza obrazowa znajduje coraz to nowe dziedziny zastosowań w mechanice, fizyce, medycynie, a nawet kryminalistyce! Przedsiębiorstwa polskie prezentowały szeroką gamę rozwiązań. Proste i stosunkowo tanie (8 mln zł bez komputera), oparte były na polskiej kamerze TV użytkowej typu TPK-162 oraz własnej konstrukcji karcie 7-bitowego przetwornika A/D wraz z kartą sterownika monitorów (128 poziomów szarości lub pseudokolorów, rozdzielczość 256x256 punktów - oferta firmy Omega z Łodzi). Złożone systemy analizy obrazów kolorowych w czasie rzeczywistym oparte na kamerze wideo firmy Panasonic oraz karcie cyfrowego przetwarzania obrazu (prod. tajwańskiej) o rozdzielczości 512 x 512 punktów, 32 poziomach jasności każdego z kolorów podstawowych (R,G,B) i szybkości akwizycji 25 obrazów na sekundę za jedyne 18 mln zł (bez komputera) oferowało przedsiębiorstwo "Sirpol" z Sieradza.

Inną ofertę z tej dziedziny przedstawiła Politechnika Warszawska pokazując na targach system Gamma-PW, który - w przeciwieństwie do poprzednich opracowań - jest systemem wąskospecjalizowanym w dziedzinie analizy obrazów scyntygraficznych w medycynie nuklearnej. Wyposażony jest on w specjalistyczne oprogramowanie przeznaczone do obliczania z uzyskanych obrazów charakterystycznych parametrów układu krążenia badanego pacjenta. System ten może pracować zarówno z komputerem IBM PC XT/AT (wersja z 1 monitorem graficznym), jak i z MERĄ 660 pod nadzorem systemu op. RT-11 (wersja z 2 monitorami graficznymi). Oprogramowanie specjalistyczne jest w pełni zgodne


1



z oprogramowaniem medycznym dla komputerów firmy DEC.

* * *

W dziedzinie analizy przebiegów EEG znanemu sprzed 2 lat, a obecnie udoskonalonemu programowi "Neuroscan" gliwickiej firmy "Proster" przybyli konkurenci tacy jak "EEG-Comers" warszawskiej spółki "Comers-Electronic", czy też "BrainStar" firmy "Uni-Med" z Warszawy. Podobny program, ale dla Atari serii ST oferuje białostocki oddział Centralnego Ośrodka Techniki Medycznej w Warszawie. Przedstawiciel "Prosteru" zapewnia, że w tej dziedzinie polscy programiści są w czołówce światowej, a ich programy budzą zainteresowanie w krajach Europy Zachodniej. Potwierdzałby to fakt zawarcia pomiędzy tą firmą a węgierskim potentatem w przemyśle elektroniki medycznej - Zakładami "Medicor" - kontraktu na sprzedaż aparatów EEG z programem "Neuroscan".

Innym ciekawym programem jest "SAAS". Przeznaczony jest on przede wszystkim do badań odporności osób na zmęczenie w pracy zmianowej i długotrwałej. Jest to niesłychanie ważne przy doborze osób do pracy w takich zawodach jak lotnik, maszynista kolejowy czy kierowca. Stosuje się go także do diagnostyki osób cierpiących na zaburzenia snu oraz w badaniach wpływu leków na aktywność mózgu podczas snu. Program korzysta m.in. z EKG i EEG dostarczanych do szybkiego 12-bitowego przetwornika A/D. Koszt programu z przetwornikiem i instalacją systemu wynosi 7-8 mln zł.

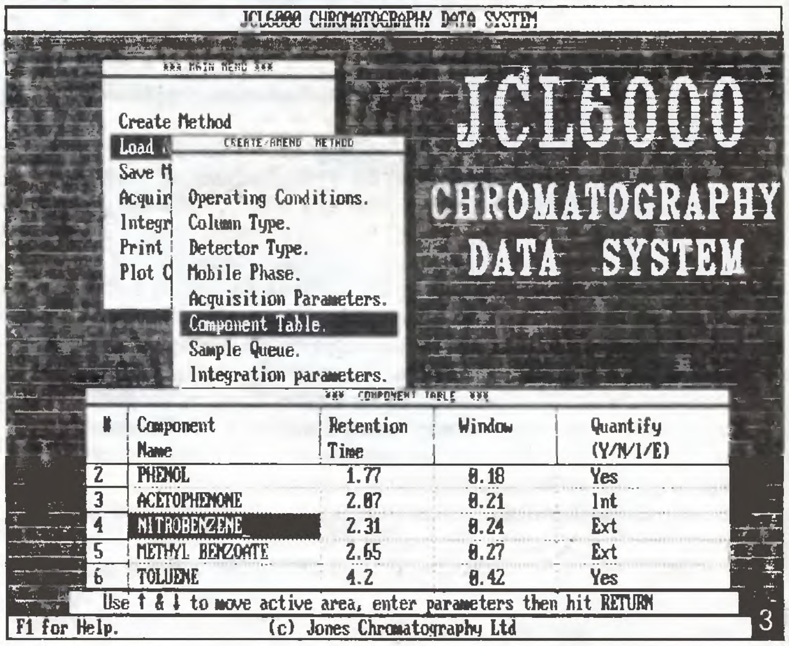
Wśród programów do analizy EKG wyróżnić można EKG Wysokiego Wzmocnienia firmy "Mercomp" z Warszawy, w którym zastosowano metody selekcjonowania użytecznego sygnału EKG z szumów oraz "StressTest" "Prosteru", w którym zastosowano metodę komputerowego uśredniania w celu usunięcia z zapisu EKG zakłóceń spowodowanych ruchem chorego podczas badania. Bez zastosowania komputera zapis takiego badania jest trudny do interpretacji. Program Interpretacji EKG przedstawiły też Zakłady Techniki Medycznej "Temed" z Zabrza. Współpracuje on z przystawką EKG, która cyfrowo koduje sygnał EKG i przesyła go do komputera IBM PC XT/AT poprzez łącze szeregowe RS 232C.

* * *

Wiele firm oferowało programy baz danych obsługujących różne dziedziny związane z medycyną. Dało się zaobserwować programy dotyczące ruchu chorych, aptek czy też stacji krwiodawstwa, a nawet programy automatycznej rejestracji wyników sekcji zwłok. Wszystkie napisano z przeznaczeniem dla IBM PC XT/AT w standardowej konfiguracji. W przeważającej większości oprogramowanie to napisane zostało w Pascalu lub dBase III w mutacjach Clipper lub FoxBase. Były również programy współpracujące z minikomputerem SM-4, np. system ADMED z ZETO Wrocław składający się z 8 modułów kompleksowo obsługujących szpital. Program ten napisano w oparciu o sieciową bazę danych BEEBASE i język Cobol. Pracuje on pod kontrolą systemu operacyjnego DOS RW V2. Jest to system wielodostępny i w związku z tym możliwa jest praca z wielu terminali równocześnie. Można było także spotkać programy pisane w języku dBase II i przeznaczone dla wszystkich mikrokomputerów pracujących pod kontrolą systemu operacyjnego CP/M 2.2. Ciekawostką było pojawienie się kilku programów ekspertowych, np. "Infarctest" służący do oceny zagrożenia zawałem serca, czy też "Nadciśnienie" - do kontroli i leczenia nadciśnienia tętniczego krwi (oba programy firmy "Refleks").

* * *

W grupie innych zastosowań mikroprocesorów i mikrokomputerów wyróżnić można system o nazwie "Automat do określania lekowrażliwości bakterii AMB-88" produkcji Zakładu Doświadczalnego Techniki Medycznej z Łodzi. Urządzenie, oparte na procesorze 8080, umożliwia więcej niż 4-krotne skrócenie czasu otrzymania antybiogramu (badania oporności bakterii na antybiotyki) dla 13 leków równocześnie oraz pełne zautomatyzowanie procesu. Umożliwia to wydatne zwiększenie szybkości i bezpieczeństwa wykonywania badań tego typu. Innym przydatnym urządzeniem jest system mikrokomputerowy do masowych badań układu oddechowego autorstwa Centralnego Ośrodka Techniki Medycznej z Warszawy. System ten składa się z sondy pomiarowej połączonej z kartą przetwornika A/D oraz odpowiedniego oprogramowania dla IBM PC XT. Jego upowszechnienie umożliwiłoby przeprowadzanie rutynowych badań układu oddechowego w dużych populacjach ludzkich narażonych na zwiększone ryzyko zapadalności na choroby płuc.



3

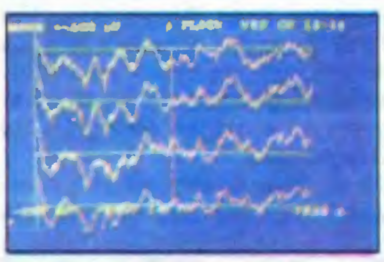
Opis ilustracji:

1. MIPRON - koncernu KONTRON Instruments służy do przetwarzania obrazów w radiologii medycznej. Dane techniczne: szybkość procesora graficznego 10 mln operacji na sekundę, szybka półprzewodnikowa pamięć obrazu 4-64 MB, rozdzielczość obrazu 4096x4096 punktów.

HP 1090 M - w pełni skomputeryzowany chromatograf cieczowy produkcji firmy Hewlett-Packard.

3. Wygląd ekranu IBM PC XT/AT sterującego skomputeryzowanym chromatografem JCL 6000 firmy JONES Chromatografy.


2





\Kurier\

Wyrwać się z getta kalectwa

W dniach 15 - 16 kwietnia br. odbyło się w Krakowie spotkanie nauczycieli szkół specjalnych stosujących w swej pracy sprzęt komputerowy. Stało się ono dla nas okazją do przeprowadzenia dyskusji poświęconej zagadnieniom wykorzystania mikrokomputerów w pracy z dziećmi specjalnej troski. Uczestniczyli w niej: Krzysztof Święcicki z Ministerstwa Edukacji Narodowej, Franciszek Chudzio - dyrektor Szkoły Specjalnej Nr 131 w Krakowie, Hanna Tkacz - dyrektor Zasadniczej Zawodowej Szkoły Specjalnej Nr 51 w Warszawie, Tomasz Ciesielski - nauczyciel w Szkole Specjalnej Nr 131 w Krakowie oraz Stefan Świszczowski - pracownik naukowy Politechniki Krakowskiej.

Red.: - *Czy widzicie Państwo szanse na rozpowszechnienie się mikrokomputerów w szkołach specjalnych?*

Krzysztof Święcicki: - Oczywiście. Uważam, że komputer w szkole specjalnej pojawić się musiał. Pozostaje pytanie czy stworzy nowe szanse przed edukacją. Ta konferencja i doświadczenia szkoły, która ją zorganizowała, świadczą, że w szkolnictwie specjalnym komputer może stanowić szansę nawiązania kontaktu z dzieckiem upośledzonym.

Red.: - *Upośledzonym, czyli jakim?*

Franciszek Chudzio: - W naszej szkole uczą się dzieci o obniżonej sprawności intelektualnej, upośledzone umysłowo w stopniu lekkim, z zaburzeniami emocjonalnymi, psychiatrycznymi, dysfunkcją ruchu, np. ręki itd. Szukając nowych form oddziaływań na nie kilka lat temu podjęliśmy próbę wykorzystania sprzętu komputerowego. Przez dobry przypadek trafił do nas doktor Świszczowski. Początkowo wykorzystywał na zajęciach własny sprzęt, z czasem zorganizował grupę osób, która podarowała mikrokomputer szkole. Warunek był taki, że sprzęt będzie służył dzieciom. Znaleźli się młodzi nauczyciele, między innymi mgr Ciesielski i mgr Pałyga. Chcieli zrobić więcej, niż wymagał tego program nauczania, dając dzieciom szansę rozwoju intelektualnego, wyrównywania pewnych zaburzeń emocjonalnych. Przekonaliśmy się, że umiejętne posługiwanie się komputerem może wielokrotnie wzmocnić własne oddziaływanie dydaktyczne na dzieci.

Red.: - *Czy Państwo zauważyli pozytywny wpływ tego sprzętu na dzieci, które z racji swojego stanu zdrowia nie zawsze mogą w normalny sposób pobierać naukę?*

Franciszek Chudzio: - Na każde dziecko uczestniczące w zajęciach komputerowych ćwiczenia te wpływają inaczej. Taki na przykład Adam, który jest dzieckiem nadpobudliwym, gdy siada do komputera, natychmiast się uspokaja, potrafi się dłużej skoncentrować. Dzieci, które po kilku minutach lekcji nie są w stanie usiedzieć na miejscu, podczas pracy z komputerem potrafią się skupić przez godzinę i dłużej. Doskonałym przykładem może być siedemnastolatek, który trafił do nas po ukończeniu piątej klasy. Cierpiał na kilka schorzeń ograniczających jego zdolność nauczenia się czegokolwiek, a zwłaszcza matematyki. W tym wypadku wielokrotne powtarzanie zajęć z komputerem doprowadziło do tego, że chłopiec nauczył się nawet tabliczki mnożenia, co wcześniej wydawało się absolutnie niemożliwe. Okazało się, że nieograniczona cierpliwość maszyny dała w tym wypadku efekty trudne do uzyskania w konwencjonalnym nauczaniu.

Hanna Tkacz: - Efekty widoczne są czasami od razu po pierwszej lekcji. Dla dzieci, które nie myślą abstrakcyjnie, bo jest to poza sferą ich możliwości intelektualnych, wykorzystujemy programy, które pozwalają im kojarzyć i zrozumieć zagadnienie, np. program "ułamki".

Franciszek Chudzio: - Początkowo dysponowaliśmy kilkoma programami w języku angielskim otrzymanymi wraz z mikrokomputerem. W miarę upływu czasu dopasowaliśmy je do naszych potrzeb. Robili to sami nauczyciele w trakcie poznawania sprzętu. Najbardziej zaangażowany, Tomasz Ciesielski, zdecydował się na podjęcie podyplomowych studiów w dziedzinie informatyki. Powstał zespół ludzi związanych ze szkołą, którzy pomagali w tworzeniu programów edukacyjnych.

Red.: - *Czy nie widzieliście Państwo ujemnych skutków oddziaływania komputera na dzieci? Słyszeliśmy, że osoby chorujące na epilepsję nie powinny mieć do czynienia z szybko zmieniającym się obrazem oraz ostrymi dźwiękami.*

Tomasz Ciesielski:- Słyszałem, że kwestia migotania ekranu może mieć wpływ na pewne procesy mózgowe, ale uważam to nadal za hipotezę.

Wszystkim rodzajom upośledzenia umysłowego towarzyszą zaburzenia emocjonalne, charakteryzujące się głównie stanem pobudzenia. Te dzieci są bardziej ruchliwe. Na początku komputer jako nowość rzeczywiście wzmagał stan ich napięcia i powodował, że nie były w stanie do końca się skupić na tym, o co chodziło w procesie nauczania. W tej chwili komputer jest dla nich rzeczą normalną, tak samo jak każdy inny środek dydaktyczny. Organizacja pracy w pracowni komputerowej powoduje inny sposób reagowania, bardziej wyciszony, spokojny, poważny i odpowiedzialny. Poprzez kontakt ze sprzętem wysokiej klasy dzieci czują się dowartościowane i starają się temu sprostać. Trzeba pamiętać o tym, żeby zorganizować te zajęcia w sposób hamujący i wyciszający nastroje emocjonalne.

Red.: - *Tomasz Ciesielski jest tu osobą najbardziej zaangażowaną w tworzenie oprogramowania. Proszę nam powiedzieć, dlaczego Pan się tym zajął? Jakich rad udzieliłby Pan osobom, które chciałyby dołożyć swoją cegiełkę do upowszechnienia tej techniki, będącej często jedyną szansą dla osób niepełnosprawnych, dającej im możliwość porozumiewania się z otoczeniem, wyjścia z getta własnego kalectwa, choroby czy upośledzenia.*

Tomasz Ciesielski: - Na pewno było to zafascynowanie nową techniką. Chciałem się dowiedzieć jak "to" pracuje. Z czasem podjąłem pierwsze próby tworzenia własnych programów. Od początku zauważyłem duże możliwości sprzętu, nawet komputera typu Spectrum, jeżeli chodzi o jego zalety dydaktyczne. Doświadczenia w pracy z dziećmi zaowocowały wieloma pomysłami programów. Na początku były to trudne sprawy. Z czasem dojrzeliśmy do programów, które są wykorzystywane podczas zajęć z dziećmi upośledzonymi. Dopracowaliśmy się również pewnych ogólnych zasad, które winny przyświecać twórcom oprogramowania dydaktycznego.

Mając rozeznanie w możliwościach sprzętu, a z drugiej strony w potrzebach dydaktycznych, oraz dysponując kontaktami w środowisku związanym z komputerami w ogóle, można by było doprowadzić do powstania zespołu tworzącego takie oprogramowanie skutecznie, efektywnie i w większych ilościach. Musiałoby się to odbywać w zespołach zatrudniających nauczyciela, informatyka i psychologa.

Programy powinny być weryfikowane na miejscu przez same dzieci, tak jak się to działo u nas. Jeżeli program już istnieje w jakimś zarysie, dajemy go naszej młodzieży, patrzymy na reakcje, obserwujemy, w których momentach jest on za trudny , jak dzieci reagują, czy szata graficzna jest zbyt uboga, czy za bogata. Na bieżąco są nanoszone poprawki. Proces powstawania takiego programu jest czasem długi. Niektóre rodziły się pół roku, inne rok. Oczywiście nie oznacza to stałej pracy nad tym programem, ale proces jego dojrzewania od koncepcji do kształtu ostatecznego.

Red.: - *Czy to jest tylko kwestia oprogramowania? Czy sama klawiatura nie stanowi bariery dla dzieci niesprawnych ruchowo?*

Tomasz Ciesielski: - Szkoła nasza jest dla dzieci lekko upośledzonych, wobec czego radzą sobie one z klawiaturą. Ideałem byłoby, gdyby programiści pamiętali o ograniczaniu obsługi programu do kilku klawiszy, gdyby można było wydzielić kilka klawiszy, np. zasłaniając resztę klawiatury.

W obsługiwaniu niektórych programów bardzo pomocne są joysticki, chociaż np. wybieranie strzałek na klawiaturze pomaga korygować pewne zaburzenia - na przykład w orientacji przestrzennej czy koordynacji typu ruchowego. Dlatego można i należy to ułatwiać, ale tylko do pewnego momentu.

Franciszek Chudzio: - Dla niektórych dzieci należałoby jednak stworzyć specjalne manipulatory do porozumiewania się z komputerem. Mam tu na myśli dzieci z porażeniem mózgowym, mające

ogromne trudności w skoordynowanym posługiwaniu się kończynami.

Red.: - *Doktor Stefan Świszczowski doprowadził do zainteresowania się pracowników Szkoły Specjalnej Nr 131 w Krakowie wykorzystaniem mikrokomputera w codziennej pracy. Co spowodowało, że wraz z kilkoma osobami nabył Pan mikrokomputer, aby następnie podarować go tej szkole?*

Stefan Świszczowski: - Wydawało mi się, że jeżeli przeczytam o wykorzystaniu gdzieś mikrokomputera w szkolnictwie specjalnym i przeznaczę na to trochę swojego czasu, to doprowadzę do uzyskania podobnych efektów w naszym kraju. Okazało się, że sprawa rozwinęła się głównie dzięki zaangażowaniu nauczycieli oraz takich instytucji jak Kuratorium oraz Polskie Towarzystwo Informatyczne, które dostarczyły w późniejszym okresie następne mikrokomputery. Była to także zapobiegliwość dyrektora Chudzio, który nawiązał kontakty z Zakładem Doskonalenia Zawodowego, a ten z kolei sprowadził następne mikrokomputery oraz wyposażył szkołę w odpowiednie meble. Moja rola sprowadzała się więc do zainicjowania czegoś, co później samo rozwinęło się dzięki dużej aktywności grupy wartościowych ludzi.

Red.: - *Dotychczasowa działalność opiera się głównie na dobrej woli osób, które pracując społecznie pomagają tworzyć oprogramowanie, bardzo często nie wiedząc, w jaki sposób to robić. Są informatykami, mają dobre chęci. Czy Państwo oferujecie im swą pomoc przy tworzeniu stosownego oprogramowania?*

Hanna Tkacz: - Nie jesteśmy przygotowani do pisania programów, natomiast mamy gotowe scenariusze - tematy, jakie byśmy potrzebowali do realizacji. Szukamy dróg dojścia do społeczników, którzy by nam pomogli.

Franciszek Chudzio: - Przede wszystkim potrzebujemy informatyków, którzy potrafiliby stworzyć programy na podstawie wspomnianych scenariuszy. Konieczna byłaby też współpraca specjalistów z innych dziedzin.

Red.: - *Proszę wskazać drogę postępowania osobom mającym zacięcie społecznikowskie, czy instytucjom, które chciałyby pomóc dzieciom. Ja rozumiem, że powinny się tym zająć instytucje odpowiedzialne za nasze szkolnictwo powszechne i specjalne, ale zanim do tego dojdzie, co należałoby robić i w jaki sposób pomagać?*

Tomasz Ciesielski: - Pomagać, to znaczy wspierać to, co dzieje się do tej pory oraz propagować naszą ideę stosowania komputera w szkole specjalnej. Staramy się na ten temat dużo mówić na spotkaniach i konferencjach metodycznych. Organizujemy szkolenia dla kolegów zarówno z naszej szkoły jak i innych. Prowadzimy kurs podstaw użytkowania i oprogramowania mikrokomputera na użytek nauczycieli szkół specjalnych z Krakowa. Staramy się pisać na ten temat w prasie ogólnopolskiej i lokalnej, zarówno fachowej, pedagogicznej, jak i popularnej. Zrealizowano kilka audycji radiowych, jedną telewizyjną, które były emitowane na antenie ogólnopolskiej. Robimy to oczywiście w miarę naszych skromnych możliwości, tym bardziej że przede wszystkim prowadzimy prace dydaktyczno-wychowawcze, natomiast wszystko inne, tzn. propagowanie, pomoc, kształcenie kolegów i koleżanek, jest jedynie dodatkiem. Stąd chętnie widzielibyśmy pomoc tych wszystkich, dla których nagrodą potrafi być uśmiech dziecka i satysfakcja wynikająca ze świadomości pomocy w przebijaniu się do otaczającego świata.

Red.: - *Wydaje mi się, że inicjatywa zorganizowania tego spotkania powstała w wyniku wyczerpania się lokalnych możliwości wykorzystania posiadanego sprzętu.*

Franciszek Chudzio: - Napotkaliśmy swoistą barierę w działalności i wdrażaniu mikrokomputerów, sprawdzeniu efektywności ich oprogramowania. Postanowiliśmy poszukać ludzi, którzy gdzieś, w Polsce zajmują się podobnym problemem. Wspólnie mogliśmy przedyskutować interesujące nas tematy, wymienić używane i sprawdzone programy, zorientować się w ich efektywności.

Tomasz Ciesielski: - Uważam, że fakt zebrania się przedstawicieli takich instytucji jak: Ministerstwo Edukacji Narodowej, Polskie Towarzystwo Informatyczne, redaktorów "Szkoły Specjalnej" oraz "Komputera", fundacji "Pomoc Dzieciom", lekarzy i nauczycieli szkół specjalnych, powinien w przyszłości zaowocować.

Hanna Tkacz: - Mamy nadzieję, że nasze krakowskie spotkanie będzie motorem. Zaczynamy mówić o istnieniu problemu. Jego rozwiązanie wymagać jednak będzie pomocy administracji oświatowej.

Red.: - *Pozostaje nam więc mieć nadzieję, że efektem tego spotkania będzie również zwiększenie efektywności działania odpowiednich instytucji. Dziękujemy Państwu za udział w dyskusji.*

Dyskusję prowadził: **Włodzimierz Banaszak**

\Kurier\

Wojciech Olejniczak

Dostęp

- *Osoba niepełnosprawna nie musi mieć komputera wysokiej klasy: najważniejszy jest dostęp do niego* - twierdzi Sławomir Besowski, dyrektor biura usług komputerowych "APCO" sp. z o.o.

Dyrektor Besowski nie ma fotela odpowiedniego do zajmowanego stanowiska, lecz wózek inwalidzki. O możliwościach zastosowania techniki komputerowej dla osób niepełnosprawnych może powiedzieć wiele. Nie tylko z autopsji albo dlatego, że kieruje firmą komputerową - pokazuje literaturę, filmy wideo i ma ochotę na rozmowy.

- *Ta technika może w dużym stopniu pomóc osobom z porażeniem mózgowym* - mówi. *Przypomina film emitowany w naszej telewizji. To, że spikerka zapowiedziała go dla osób o silnych nerwach, świadczy o stosunku u nas do osób niepełnosprawnych* - dodaje. – *A pokazano nie jakiś tam thiller z okrwawionymi ofiarami wampira, tylko ośrodek rehabilitacji, uczący najdotkliwiej upośledzone osoby kontaktu ze światem przy pomocy komputera. Zgrozę u telewidzów mógł wywołać na przykład niemowa zamawiający pizzę przez telefon. Pominąwszy pizzę i to, że można przez telefon. Przykład był jasny: ci ludzie mogą pełniej żyć dzięki technice komputerowej. Komputer może, nie wdając się w techniczne szczegóły, przy użyciu odpowiedniego manipulatora korzystającego z ruchu ręki, nogi czy nawet gałki oka przekształcić przesyłane sygnały na mowę.*

Mało tego - dyrektor przedstawia film na wideo, na którym sparaliżowany człowiek porusza się przy pomocy innych, sprawnych mięśni. Świadomie poruszając barkiem lub szyją przenosi sygnały (elektrody w mięśniach) via komputer, pomijając uszkodzone odcinki rdzenia kręgowego, do mięśni pozbawionych sterowania. Okazuje się, że tego rodzaju zabieg daje efekty lecznicze, powodując niekiedy powstanie innych dróg nerwowych. Mózg uczy się kierowania mięśniami w inny niż ukształtowany przez naturę sposób i ingerencja z zewnątrz przestaje być konieczna.

- *Technika pozwala na rozwiązywanie problemów niepełnosprawnych, ale najważniejszy jest dostęp do niej* - powtarza dyrektor Besowski. Z tym postępem jest tak, jak z przystosowaniem człowieka po wypadku do nowej sytuacji. Sławomir Besowski obronił pracę magisterską (symulacja pracy silnika turbinowego przy użyciu komputera) na piątkę i zatrudnił się w Instytucie Lotnictwa. Może teraz byłby po doktoracie, gdyby nie błąd w czasie sterowania lotnią. Latanie skończyło się poważnym - jak określają lekarze - czterokończynowym uszkodzeniem rdzenia w odcinku szyjnym, czyli po prostu: sparaliżowaniem nóg i rąk.

- *W naszym systemie społecznym wbrew niejednokrotnym deklaracjom o opiece, gwarantowaniu godziwej egzystencji, ograniczenia ruchowe uniemożliwiają dostęp do odpowiedniej, interesującej pracy, czyli, w rezultacie, normalnego życia* - mówi Sławomir Besowski. W przypadku osób z wyższym wykształceniem to jest tragedia - niepełnosprawni mogą (spółdzielnie pracy inwalidów) wykonywać tylko niskokwalifikowane czynności, nie wymagające odwoływania się do predyspozycji intelektualnych. Powstało spółdzielcze getto inwalidów, gdzie o przydatności decyduje wykonywanie normy przy "durnej" maszynie wymagającej przyciśnięcia guzika. Można zostać w domu, oddać się rodzinnej opiece i państwowej rencie. W konsekwencji wywołuje to bierność, a bywa, że ograniczenie sprawności ruchowej, ba nawet psychicznej, bo wszystko to się ze sobą wiąże.

Ośrodki rehabilitacji ludzie opuszczają w dobrej formie. Cóż z tego, gdy nie mogą zająć się swoimi sprawami, napotykając bariery nie do przebycia? Głupia sprawa: schody. Gdy dyrektor Besowski spróbował kontynuować pracę w swoim instytucie, stykał się ze stopniami na co dzień. Musiał korzystać z pomocy innych, umawiać się na godzinę, żeby przenieść wózek z dołu na górę i odwrotnie. Rodziło się naturalne w tej sytuacji poczucie zależności, skrępowania, że wywołuje się zamieszanie. U nas praktycznie nie ma pojazdów, wygodnych wind, szerokich drzwi... Kiedy na wózku poszukuje się dostępu do komputera, człowiek poznaje smak infrastruktury pozbawionej odrobiny myśli o niesprawnych ludziach.

Kolega lotniarz, jeśli można tak powiedzieć, bo wiekiem i stopniem naukowym o wiele bardziej zaawansowany, pomógł dwustoma dolarami na

12

11 <

Sinclaira, a to dało okazję do zrobienia paru zleceń i co ważniejsze - wywołało refleksję: a może to jest szansa? Znalazł się inny kolega gotów wspomóc gotówką i otwarty na propozycję - Sinclair został sprzedany, a na jego miejsce pojawił się IBM PC. Do dziś pracuje na użytek spółki, początkowo towarzyskiej, z czasem formalnej.

I tu można dopisać optymistyczną pointę: wózek nie przeszkodził Sławomirowi Besowskiemu zostać dyrektorem Biura Usług Komputerowych...

- *Dopóki niepełnosprawny jest bierny, oczekuje pomocy - to jest normalne. Gdy natomiast chce wziąć sprawy w swoje ręce, napotyka co najmniej niezrozumienie* - zauważa dyrektor Besowski. - *Na przykład gdy przedstawiam się przez telefon jako dyrektor firmy, jest jeszcze w porządku. Ale gdy dochodzi do spotkania twarzą w twarz... zaczyna się mnie traktować jak dziecko, albo człowieka niespełna rozumu, a w najlepszym przypadku osobę budzącą litość.*

Tymczasem nie litość jest potrzebna, a przede wszystkim lokal przystosowany dla ludzi niepełnosprawnych. Na razie biuro "APCO" mieści się w pokoju o powierzchni 24 m², w prywatnym mieszkaniu dyrektora na II piętrze. Firma zatrudnia jeszcze dwie niepełnosprawne osoby, ale więcej nie przyjmie. Po prostu: jak się zbierze troje na wózkach, by wspólnie coś zrobić, w pokoju nie można się ruszyć.

- *A moglibyśmy dać wielu osobom o podobnych dolegliwościach szansę na spełnienie zawodowych ambicji. Pracy jest mnóstwo* - deklaruje dyrektor Besowski.

Następny przykład z wideo dyrektora: człowiek ze skoordynowanymi ruchami kilku palców stopy, pozbawiony mowy. Przy pomocy rodziny i dzięki różnym elektronicznym udogodnieniom zrobił dwa fakultety i doktorat. Komunikował się z otoczeniem poprzez klawiaturę komputera.

- *Stworzyliśmy na bazie Commodora 64 system, który spełnia podobne funkcje* - mówi dyrektor. Na monitorze widoczny jest alfabet, sterując kursorem (do tego może służyć dowolny manipulator, począwszy od zwykłego joysticka po bardziej skomplikowane, indywidualne urządzenia) układa się słowo, ze słów zdania i tak dalej, widoczne na dole ekranu w postaci tekstu. Poza tym najprostsze teksty (otwórz okno, podaj książkę itp.) można zakodować w postaci zestawu standardowych poleceń.

- *Chcieliśmy zainteresować tą sprawą ministerstwo zdrowia, ale oglądający naszą propozycję urzędnik stwierdził, że widzi jej ewentualne zastosowanie w szpitalach albo ośrodkach rehabilitacyjnych. A przecież jest to sprzęt osobisty, potrzebny na co dzień, tak jak aparat słuchowy. Całą rzecz można przecież zrobić na specjalizowanym komputerze, nie zawsze potrzebna jest klawiatura, w przypadku indywidualnych potrzeb jest możliwe zbudowanie odpowiedniego manipulatora na zamówienie. Można też wykorzystać program syntezy mowy.*

System został wystawiony na targach "Rzemiosło - medycynie". Również bezskutecznie. Chociaż zgłosił się do "APCO" niewidomy z resztkami słuchu, który przy pracy z komputerem posługiwał się praktycznie dotykiem (klawiaturę znając na pamięć). Pisał programy, ale ich korektę musiał dokonywać przy pomocy innych osób.

- *Układ, dzięki któremu mógłby słyszeć pisany tekst, byłby dla niego fantastyczny* - mówi dyrektor Besowski. - *Można też w końcu zrobić coś w rodzaju monitora Braille'a, czytnika pokrytego siecią igieł, choćby z głowicy drukarki mozaikowej.*

Każdy pomysł kosztuje. System zbudowany na Commodorze to wydatek 300 tysięcy złotych. Niewiele rodzin na to stać. W kraju, w którym brakuje pieniędzy na leki, trudno spodziewać się objęcia tego typu sprzętu ubezpieczeniami społecznymi.

Szef "APCO" oponuje: - *W Stanach Zjednoczonych obliczono, że każdy dolar zainwestowany w aktywizację społeczno - zawodową niepełnosprawnych przynosi dziesięć dolarów zysku. W Polsce jedna złotówka daje nawet dwadzieścia. Nie wszystko możliwe jest tylko w Ameryce, ta technika dociera do nas w coraz większym stopniu i zbudowanie u nas różnych udogodnień elektronicznych dla osób poszkodowanych jest technicznie możliwe.*

Ale cóż, u nas wszyscy są zajęci ratowaniem życia. Mało kto interesuje się tymi, którzy przeżyli - dodaje.



Kurier

Stanisław Jakubowski

Komputery pomogą niewidomym?!

Jedną z największych trudności, jaką niewidomi muszą pokonać w czasie nauki i wykonywania pracy, jest ograniczony dostęp do informacji wizualnej, a w szczególności pisanej.

Problemy zaczynają się już w szkole podstawowej, gdzie tak potrzebny środek komunikacji między nauczycielem a uczniami, jakim jest tablica, to dla niewidomych bezużyteczny przedmiot.

Kolejną trudność stanowi niedobór podręczników szkolnych. Są one wydawane przez jedyną w kraju drukarnię Polskiego Związku Niewidomych, która pracuje na mocno już wyeksploatowanych urządzeniach. Dlatego podręczniki i lektury szkolne ukazują się z dużym opóźnieniem i pokrywają w pełni zapotrzebowanie zaledwie na poziomie szkoły podstawowej. Należy w tym miejscu uświadomić Czytelnikowi, że książki dla niewidomych to niespełna 1% wszystkich edycji, jakie ukazują się na naszym rynku księgarskim.

Obok dostępu do informacji równie ważnym problemem dla niewidomych jest prezentacja własnych tekstów. Można go częściowo rozwiązać poprzez opanowanie do perfekcji klawiatury zwykłej maszyny do pisania. Pracując jednak w taki sposób niewidomy pozbawiony jest możliwości odczytania napisanego przez siebie tekstu oraz przeprowadzenia jego korekty.

Pismo

Powyższe trudności można rozwiązać w oparciu o najnowsze zdobycze techniki rozszerzające znacznie zakres zastosowań pisma niewidomych. Wynalezione w 1824 roku przez 15-letniego Francuza, Ludwika Braille'a, pismo stało się wielkim dobrodziejstwem dla wzroku inwalidów. Podstawowym elementem pisma Braille'a jest prostokątna macierz sześciu wypukłych punktów, ułożonych w trzech wierszach i dwóch kolumnach. Macierz ta zwana jest po prostu sześciopunktem. Każdą literę można w alfabecie brajla uzyskać wybierając odpowiednią kombinację kropek w ramach sześciopunktu. Należy przy tym podkreślić pewną istotną cechę pisma niewidomych, a mianowicie fakt, że zbiór znaków alfabetu brajlowskiego (uwzględniając spację) liczy zaledwie 64 elementy. Dlatego wiele znaków graficznych pisma ludzi widzących zastępuje się tu sekwencjami dwóch i więcej symboli. Swoje pismo niewidomi odczytują poprzez dotyk. Za przeciętną prędkość czytania można uznać 100 słów na minutę, co odpowiada mniej więcej jednej trzeciej prędkości czytania w przypadku osób posługujących się wzrokiem.

Już tylko to krótkie omówienie najistotniejszych cech pisma brajla wystarczy, aby Czytelnik zdał sobie sprawę z jego odmienności i zrozumiał, dlaczego jest ono prawie nieznane ogółowi społeczeństwa.

Notesy i czytniki

Ostatnie dwudziestolecie przyniosło szereg wynalazków, które wprawdzie nie stwarzają niewidomym możliwości widzenia, ale rozwiązują kolejne problemy ich życia i pracy zawodowej. Powstały one dzięki szybkiemu rozwojowi elektroniki.

Jako pierwsze pojawiły się przenośne magnetofony, pociągając za sobą powstanie nowej formy wydawnictw dla niewidomych, jaką jest nagrywanie książek na kasetach. Małe magnetofony służą również niewidomym do nagrywania wykładów, a także do szybkiego utrwalania w formie wypowiedzi - własnych notatek bądź opracowań.

W ubiegłym dziesięcioleciu ukazały się aparaty do czytania dla niewidomych, a mianowicie Optacon i Stereotoner. Pierwszy z nich oddaje wierny rysunek danej litery za pośrednictwem wibrujących pręcików, które wyczuwane są za pomocą dotyku. Stereotoner zaś prezentuje dany znak graficzny w formie ciągu akordów, z których każdy jest kombinacją w ramach dziesięciu tonów o zróżnicowanej wysokości.

Chociaż liczba użytkowników Optaconu wynosi w całym świecie ok. 15000, to wynalazek ten nie przyjął się powszechnie. Powodem jest nie tylko zawrotna cena (5000 dolarów USA), ale chyba w jeszcze większym stopniu - wysokie wymagania psychofizyczne stawiane niewidomemu użytkownikowi oraz stosunkowo mała szybkość czytania. Nawet najsprawniejsi użytkownicy aparatu rzadko czytają więcej niż 70 słów na minutę.

Stereotoner natomiast był wynalazkiem efemerycznym. Zdobył niewielu zwolenników, gdyż osoby posługujące się nim musiały mieć absolutny słuch muzyczny. Tempo czytania było jeszcze mniejsze niż w przypadku Optaconu (co najwyżej 40 słów na minutę), toteż po kilku latach słuch o Stereotonerze zaginął.

W końcu lat siedemdziesiątych w USA pojawiło się urządzenie spełniające marzenia wszystkich niewidomych: maszyna odczytująca materiały drukowane. Jest to czytnik optyczny pisma wyposażony w mowę syntetyczną dla języka angielskiego. Nosi on nazwę Kurzweil Talking Machine (KTM) i kosztuje obecnie 25 000 dolarów. Wysoka cena, nawet na wa-

runki amerykańskie, oraz znaczne ograniczenia co do rodzaju druku i formatu czytanego automatycznie tekstu sprawiły, że KTM wykorzystywana jest na terenie krajów anglojęzycznych jedynie w instytucjach działających na rzecz niewidomych, a nie przez indywidualnych użytkowników.

Znacznie większe sukcesy zdobył inny wynalazek, który w dużej mierze umożliwił osobom posiadającym resztki wzroku, tj. ludziom słabowidzącym, czytanie większości materiałów pisanych. Mowa tu o tzw. powiększalniku telewizyjnym. Zasada jego działania polega na rejestrowaniu tekstu przez kamerę wyposażoną w układ powiększający o zmiennej ogniskowej i wyświetlaniu go na ekranie telewizyjnym. Ponieważ tak działające urządzenie powiększa daną literę kilkanaście razy, korzystać mogą z niego osoby nawet bardzo źle widzące. Liczba słabowidzących użytkowników powiększalnika telewizyjnego na całym świecie z pewnością przekroczyła już 100 000. Jego produkcja - w ograniczonym zakresie - podtrzymywana jest w naszym kraju, a liczba właścicieli niebawem osiągnie wartość 500.

Komunikowanie się niewidomego z komputerem

Techniki te nie rozwiązują problemów osób całkowicie niewidomych, dla których księgozbiory zwykłych bibliotek pozostają wciąż niedostępne. Komputery osobiste zrewolucjonizowały już wiele dziedzin działalności ludzkiej, a zwłaszcza te, które związane są z przetwarzaniem, gromadzeniem i przesyłaniem informacji. Wielu niewidomych stawia sobie pytanie, czy informatyzacja licznych dziedzin naszego życia stanie się również ich udziałem.

Aby odpowiedzieć na to pytanie, zastanówmy się, czy jest w ogóle możliwa komunikacja osoby niewidomej z komputerem.

Dla ludzi pozbawionych wzroku konieczne są urządzenia, które umożliwiają obustronny przepływ informacji w relacji komputer-człowiek przy wykorzystaniu środków niewizualnych. W jednej z wersji stanowisko dla niewidomego informatyka wyposażone jest w dwa dodatkowe urządzenia. Pierwsze z nich można chyba najtrafniej określić mianem monitora brajlowskiego, gdyż podobnie jak w zwykłym monitorze ekranowym ukazują się wszystkie informacje tekstowe. Prezentowane są one jako znaki brajlowskie. Mogą być odczytywane za pomocą dotyku dzięki temu, że poszczególne punkty brajlowskie formowane są za pomocą cienkich zaokrąglonych na górze bolców, które wysuwają się nieco ponad obudowę urządzenia. Aby zbiór znaków brajlowskich mógł w pełni oddać wszystkie znaki kodu ASCII, trzeba było sześciopunkt zastąpić ośmiopunktem. Ośmiopunkt taki jest często nazywany modułem brajlowskim. W ten sposób liczba wszystkich znaków brajlowskich stała się równa ilości znaków, jaką posługują się informatycy widzący. W zmodyfikowanym bowiem alfabecie brajla wynosi ona 256 elementów.

Na podkreślenie zasługuje fakt, że informacje zapisywane i odczytywane przez wyspecjalizowane urządzenia dla niewidomych nie występują w pamięci komputera jako kod brajla. Zapis ten utrwalany jest w kodzie ASCII, a w postaci znaków brajlowskich przedstawiany jest dopiero w urządzeniach pomocniczych. Występuje tu pełna analogia do sposobu korzystania z komputera przez ludzi posługujących się wzrokiem. Zauważmy, że informacje tekstowe wyświetlane na ekranie komputerów są również przechowywane w pamięci jako poszczególne znaki kodu ASCII, a na postać czytelną dla oka ludzkiego zamieniane są dopiero dzięki "elektronice" monitora. Opisane wyżej urządzenia umożliwiają niewidomemu tworzenie i odczytywanie informacji dostępnych dla każdego. Tym samym rozwiązany został problem o kapitalnym znaczeniu dla niewidomych, a mianowicie konwersja pisma brajla na pismo ludzi widzących i na odwrót. Innymi słowy, jeśli te dwa rodzaje pisma utrwalone są w postaci cyfrowej, a nie graficznej, to teoretycznie nie istnieją już dla niewidomych przeszkody w dostępie do informacji pisanej.

Monitory, klawiatury i drukarki brajlowskie

Powstało już kilka firm produkujących monitory brajlowskie o różnej długości "linijki", tj. ilości modułów. Na przykład w amerykańskim urządzeniu Versabraille wynosi ona 20. Dla niewidomych najwygodniejsze jednak są urządzenia, które mogą prezentować przynajmniej jeden wiersz monitora ekranowego, czyli 80 znaków. Wymagania te spełniają urządzenia zachodnioniemieckie Brailloterm, Braille-Window oraz cała rodzina monitorów o wspólnej nazwie Braillex. Ich cena przekracza niestety 7 000 dolarów USA. Łatwo zatem domyślić się, dlaczego nie są produkowane monitory brajlowskie mogące prezentować jednocześnie cały ekran komputera. Aby niewidomy mógł jednak korzystać z konwencjonalnego ekranu, urządzenia brajlowskie wyposaża się w dodatkowe wskaźniki, podające numer obserwowanego wiersza i numer pozycji kursora w wierszu. Sam kursor przedstawiony jest w brajlu jako ośmiopunkt, tzn. prostokąt utworzony przez osiem bolców. Na przykład Braille-Window, dzięki zainstalowaniu wewnątrz komputera wyspecjalizowanej karty, korzysta z pamięci ekranu i może, oprócz samego tekstu, pokazywać niektóre atrybuty. Nie uwzględnia jednak kolorów. Poważnym ograniczeniem znanych monitorów brajlowskich jest fakt, że mogą one pracować wyłącznie w trybie tekstowym.

Drugim urządzeniem przydatnym niewidomemu do współpracy z komputerem jest klawiatura brajlowska. Jej obsługa jest bardzo prosta, gdyż do napisania dowolnego znaku wystarczy osiem klawiszy.

Na rynku zachodnim istnieje również kilkanaście rodzajów drukarek, które sterowane z komputera wytłaczają punkty brajlowskie na papierze przypominającym konwencjonalną składankę. Wśród tych urządzeń największe zainteresowanie budzi drukarka norweska Braillo 400, która tłoczy wypukłe punkty z szybkością 400 znaków na sekundę i to po obu stronach papieru.

Syntezery mowy

Inną możliwość współpracy z komputerem stwarza dla niewidomych wykorzystanie syntezatorów mowy. Urządzenia generujące mowę syntetyczną, zwane popularnie wyjściem głosowym (ang. *voice output*) stają się coraz tańsze i doskonalsze. Na przykład wyjście głosowe brytyjskiej firmy Dolphin Systems for the Disabled zostało wyposażone w odpowiedni edytor tekstów i może "wypowiadać" informacje z uwzględnieniem nawet kolorów, w jakich są wyświetlane na ekranie komputera. Jego cena wynosi 250 funtów. Urządzenia z wyjściem głosowym z powodu swej niskiej ceny są bardziej rozpowszechnione wśród niewidomych informatyków niż monitory brajlowskie.

W naszym kraju prowadzone są w kilku ośrodkach zaawansowane badania nad polską mową syntetyczną. Najbardziej chyba znane są wyniki Instytutu Biocybernetyki i Inżynierii Biomedycznej PAN osiągane przez mgra inż. Konrada Łukaszewicza. Wiele nadziei niewidomi entuzjaści komputerów wiążą także z inną wersją wyjścia głosowego opracowywanego przez mgra inż. Jana Grębeckiego i młodego, niewidomego informatyka - mgra Marka Kalbarczyka. Wydaje się, że już tylko miesiące dzielą nas od ukazania się urządzenia z wyjściem głosowym w języku polskim, osadzonym w wybranym edytorze tekstowym, które umożliwi inwalidom wzroku kontakt z komputerem.

Jeśli chodzi o informatyków słabowidzących, to istnieje już wiele rozwiązań sprzętowo-programowych, które dają na ekranie komputera wyposażonego w kartę CGA znaki o kilkakrotnym powiększeniu. Ostatnio pojawiły się również programy, które powiększają litery na ekranie, lecz są one niestety opracowane dla oryginalnych komputerów IBM lub ściśle z nimi zgodnych. Do działania tych programów niezbędne jest, aby karta CGA zawierała układy scalone o konkretnych numerach katalogowych i zastosowanie ich w naszym kraju natrafia na duże trudności.

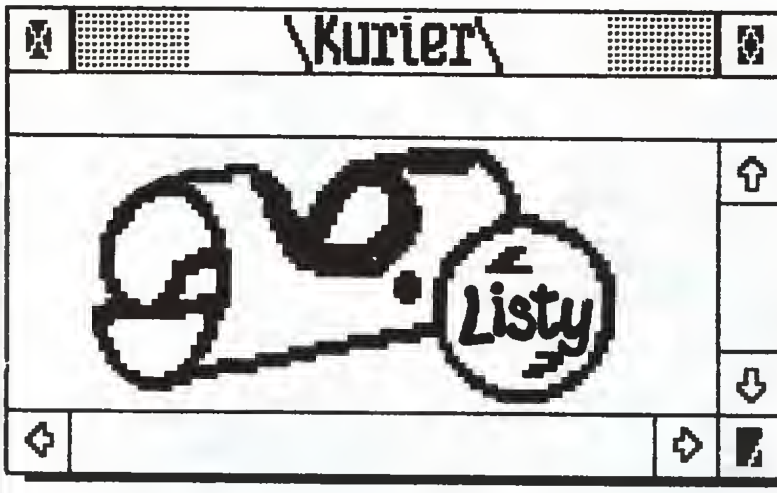
Co dalej?

Jak wynika z powyższych rozważań, komputery mogą być dla niewidomych dostępne poprzez użycie do współpracy z nimi wyspecjalizowanych urządzeń, takich jak monitory brajlowskie z klawiaturą czy wyjście głosowe oraz środków sprzętowo - programowych dających na ekranie znaki powiększone. Każda z tych metod rozwiązuje wymienione na wstępie trudności. I tak na przykład tablicą w szkole dla niewidomych mogą stać się sieci komputerowe, gdzie każda końcówka posiadałaby wyjście głosowe i klawiaturę brajlowską. Podobnie komputer wyposażony w monitor brajlowski lub mowę syntetyczną umożliwi niewidomym efektywne przygotowywanie własnych opracowań pisemnych przy zastosowaniu edytorów tekstowych. Wyprowadzenie tych tekstów na drukarkę uczyni je dostępnymi dla ogółu ludzi. Wreszcie zastosowanie komputerowych drukarek brajlowskich stwarza możliwość wykorzystania do produkcji książek dla niewidomych poligraficznych nośników cyfrowych. Rozpowszechniający się w dzisiejszym świecie cyfrowy zapis informacji stwarza dla niewidomych prawdziwą szansę w dostępie do słowa drukowanego. Wydawanie książek systemem brajla lub drukiem powiększonym na indywidualne zamówienie niewidomych i słabowidzących czytelników jest już dziś technicznie możliwe, a wdrożenie tych metod to kwestia najwyżej kilku lat.

W naszym kraju zaczęły się już ukazywać pierwsze eksperymentalne teksty brajlowskie, drukowane techniką komputerową. "Komputer" jest pierwszym w kraju czasopismem, którego wybrane artykuły wydaje się w brajlu bezpośrednio z dyskietek. Jeśli Czytelników zainteresuje to zagadnienie, redakcja "Komputera" nie odmówi łamów swego miesięcznika na zamieszczenie odrębnego artykułu o komputerowej drukarni dla niewidomych.

Doktor Stanisław Jakubowski jest niewidomym informatykiem. Biegle pisze na zwykłej klawiaturze komputera, a przygotowany tekst odczytuje za pomocą wspomnianego urządzenia o nazwie Braille-Window. Opublikowany na łamach naszego miesięcznika artykuł opracował samodzielnie korzystając z edytora WordStar i dostarczył nam go w trzech wersjach: na dyskietce, w formie komputerowego wydruku brajlowskiego oraz wydruku z drukarki mozaikowej. Tekst zawierał pełny repertuar liter polskich oraz cechowały go poprawne przenoszenia wyrazów dokonane techniką komputerową za pomocą programu, który został opracowany w drukarni brajlowskiej.





Perypetie z Venturą

W drugim numerze "Komputera" (1988) przeczytałem w jednym z artykułów ("Kronika") optymistyczną informację: "Gdy ten numer pisma dotrze do rąk Czytelników, nasza redakcja będzie testować jedną z pierwszych wersji użytkowych programu Ventura Publisher". Na marginesie, monitor ma rozdzielczość 128x960, a system DTP był prezentowany wraz ze skanerem, a nie digitizerem.

Optymizm dotyczący polskiej wersji Ventury był w pełni uzasadniony faktem wizyty przedstawicieli redakcji w warszawskim biurze firmy Rank Xerox Ltd. w listopadzie 1987, w czasie której redakcja uzyskala obietnicę przekazania jej wkrótce do testowania polskiej wersji programu Ventura Publisher 1.1. Dzisiaj (kwiecień 1988) nie pozostaje mi nic innego jak w imieniu firmy przeprosić redakcję, a także i innych, którym obiecywaliśmy zaprezentować polską wersję Ventury na targach Infosystem'88, za poważne opóźnienia.

Korzystając z okazji chciałbym przekazać trochę szczegółów dotyczących problemów tłumaczenia Ventury, które mam nadzieję usprawiedliwią powstałe opóźnienia.

Brytyjska firma Rank Xerox Ltd., będąca europejskim filarem korporacji Xerox, wraz z decyzją wejścia na rynki Europy Wschodniej z systemami mikrokomputerowymi (1987), postanowiła w ramach porozumienia z Microsoftem dostarczać podstawowe oprogramowanie w wersjach językowych poszczególnych krajów.

Dotyczy to tłumaczenia na język polski systemu operacyjnego MS-DOS 3.2 (niezbyt dobry pomysł - prywatna opinia autora), MS-Windows, MS-Word (pomysł dobry) oraz Ventury (pomysł bardzo dobry).

VENTURA Software Inc. jest nazwą amerykańskiej firmy, której głównym "produktem" jest oprogramowanie Ventura Publisher, będące prawdopodobnie standardem na rynku pulpitów wydawniczych (ang. Desktop Publishing), tutaj zapewne zaprotestują zwolennicy PageMakera z panem Stefanem Szczypką na czele.

Xerox Corporation jest oficjalnym dystrybutorem Ventury. Niestety prawa autorskie do całego pakietu posiada także Digital Research, gdyż Ventura pracuje w "otoczeniu" programu GEM oraz niewielka firma Bitstream, której własnością są czcionki, a dokładnie pliki definiujące poszczególne rodziny znaków, których Ventura zawiera kilkanaście tylko w podstawowej wersji.

Właśnie firma Bitstream nie zezwoliła na dokonanie jakichkolwiek zmian w jej części oprogramowania lub postawiła zbyt wygórowane żądania finansowe (przypuszczenie autora). Zmusiło to Rank Xerox do przygotowania niezależnie od Bitstream własnych plików definiujących czcionki.

Drugim elementem opóźniającym wprowadzenie oprogramowania, szczególnie w Polsce, są doświadczenia zbierane na wystawach komputerowych (MTP'87, Baltcom'87, Komputer'88) przez przedstawicieli firmy Rank Xerox. Oprogramowanie Ventura Publisher można oglądać pod zmienionymi nazwami, np. CAPS - Computer Aided Publishing System. Sprzedawane jest ono nielegalnie, przynajmniej z punktu widzenia firmy Rank Xerox i prawa np. brytyjskiego, oczywiście za złotówki i po stosunkowo niskiej cenie - "Easy come easy go...".

Wsławiło się tu szczególnie Przedsiębiorstwo Zagraniczne, którego nazwa sugeruje zasięg działania Globalny, przepraszam globalny, choć w rzeczywistości ograniczal się on do Berlina Zachodniego i miejscowości w okolicach Poznania.

Problem nielegalnego kopiowania oprogramowania i dystrybuowania go nawet na oficjalnych targach i imprezach w przypadku Ventury przynosi konkretne straty nie tylko dla firmy, ale przede wszystkim dla klientów. Ventura Publisher w polskiej wersji językowej będzie zawierać dodatkowe zabezpieczenia chroniące przed kopiowaniem. Koszty tego, wcale niebagatelne, pokryje częściowo polski nabywca płacąc cenę o ponad 200 dolarów wyższą od podstawowej ceny Ventury w języku angielskim.

Trzecim elementem decydującym o kolejności przygotowywania Ventury dla danego języka jest szacowane zapotrzebowanie na systemy DTP w poszczególnych krajach. Tutaj nieoczekiwanie na pierwsze miejsce wysunęły się Węgry, gdzie na majowych targach w Budapeszcie ma być zaprezentowana węgierska wersja Ventury. W tym roku mają być przygotowane jeszcze wersje: rosyjska i polska, tak więc miejmy nadzieję, że grudniowy numer "Komputera" będzie mógł zawierać stronę "złożoną" za pomocą polskiej wersji programu Ventura Publisher.

Na zakończenie wypada wskazać pozytywny element spowodowany opóźnieniem: najprawdopodobniej jesienią 1988 będzie gotowa najnowsza wersja Ventury - wersja 1.2, pracująca w otoczeniu MS-Windows (także Digital Research straci kontrolę nad Venturą). Zostanie ona automatycznie przystosowana do języka polskiego, zawierając docelowo obok charakterystycznych znaków polskich także zasady dzielenia polskich wyrazów i dokumentację w języku polskim.

Dariusz Woźniak
Rank Xerox
Warszawa

* * *

Elwro 800 Junior

Szanowna Redakcjo!

Jestem uczniem pierwszej klasy o profilu matematyczno- fizycznym VI. LO w Łodzi. Czytam Wasze pismo od pierwszego numeru. Do napisania tego listu skłoniły mnie ciągłe narzekania na niedostarczenie do testowania komputera szkolnego Elwro 800 Junior. Moja szkoła otrzymala z ministerstwa oświaty 10 komputerów Elwro 800 Junior, 10 monitorów, 3 drukarki i 3 stacje dysków. W rzeczywistości przedstawia się to mniej różowo. Jeszcze przed feriami zimowymi do szkoły przywieziono 8 komputerów, stację dysków i 8 monitorów. Lada dzień przyjechać mieli fachowcy z Unitry Serwisu, aby rozpakować nasze skarby, gdyż "samodzielne rozpakowanie komputera grozi utratą gwarancji". Czas mijał, a fachowców nie było widać. Pod koniec marca coś "drgnęło". Na korytarzu ustawiono specjalne stoliki do komputerów, a w pracowni ruszyły pełną parą roboty murarskie. 9 kwietnia zjawili się spece - dwóch ludzi, którzy byli "już" na trzecim miesiącu specjalnego kursu. Po połączeniu komputerów w sieć i sprawdzeniu poprawności działania (włączyć komputer i poczekać na napis) fachowcy "zmyli się" zostawiając jedną ciekawą uwagę: "Nie należy przejmować się za bardzo tym, co piszą w instrukcji obsługi". I rzeczywiście. Pierwsze próby zaladowania programu CP/J (Control Program for Junior ?!) zakończyły się fiaskiem. Włączanie jednocześnie klawiszy ALT, DOS daje w efekcie znak "&" zamiast obiecywanej w instrukcji komendy DOS. Dzięki wielu próbom udało nam się przebrnąć przez ten problem. Należy wpisać komendę CP/J (zastępuje ona na klawiaturze i w ROM-ie komendę MOVE), wcisnąć klawisz CR i system już "ładuje się".

Obsługa systemu jest identyczna jak systemu CP/M 2.2 (z nieoficjalnych źródeł wiem, iż CP/J i CP/M 2.2 to to samo). Można przykładowo zaladować program JUNET i (jest to jedyny sposób uzyskania w komputerach uczniowskich CP/J) uruchomić opcję "rozesłanie Systemu". Można też wysłać dowolny program do (teoretycznie) dowolnego komputera. W praktyce program JUNET w tej funkcji akceptuje tylko numer 255 (transmisja do wszystkich komputerów). Najczęściej pojawiającym się komunikatem w komputerach uczniowskich jest

"Błąd sieciowy
P owtórz Z aniechaj
I gnoruj:"

Rzadko jednak zdarza się powrócić z tego "trybu" do systemu (źle działający program?).

Do komputerów dołączono też bogatą "dyskotekę" - Turbo Pascal 3.0, Basic 5.2, omówiony przeze mnie JUNET i oczywiście dysk systemowy, na którym są m.in. SID, ZSID, MAC i M80. Do tego są jeszcze książki o obsłudze komputera, systemu, Turbo Pascala, Basica (tego z dysku oraz rozszerzenia ZX-Basic). W instrukcji obsługi kompilatora Pascala (na marginesie dodam, że jest ona identyczna z książką WNT "Turbo Pascal wersja 3.0") podano wszystkie dyrektywy kompilatora w języku angielskim, a na ekranie wszystkie komunikaty (również w trybie Spectrum i w systemie) są podawane w języku polskim. Przy okazji pewien "kwiatek". Na początku pracy program Turbo Pascal zadaje pytanie:

"Komunikaty o błędach (T/N)?"

Pomimo że w katalogu mamy zbiór o nazwie PASCAL.MSG, zdumionym oczom ukazuje się komunikat:

"PASCAL.MSG nie ma!"

Ciekawostką jest też, że program JUNET jest dostarczany w dwóch wersjach, ale druga wersja to chyba jakiś program demonstracyjny, ponieważ mimo rozłączania sieci stan jej drukowany za pomocą jednego z poleceń JUNET-a nie zmienia się. Program informatyki obejmuje kurs programowania w "Logo", jednak używanie w tym języku stacji dysków jest niemożliwe z poziomu interpretera. Niestety o magnetofony szkoła musi się postarać sama.

W sumie jednak Elwro 800 Junior jest bardzo dobrym komputerem, łatwo dostępne są polskie litery, a "teoretyczna" zgodność ze Spectrum powinna zapewnić mu wiele ciekawych programów. Piszę teoretyczna, ponieważ wydaje mi się, że zmiany w ROM-ie są zbyt duże, aby zapewnić pełną zgodność. Więcej informacji podać nie mogę, ponieważ nie znam jeszcze dobrze tego komputera.

Łączę pozdrowienia dla całej Redakcji

Robert Wal
Łódź

Redakcja nie miała możliwości bliższego zapoznania się z tym komputerem. Aby jednak nie pomijać go milczeniem, drukujemy listy Czytelników dzielących się swoimi doświadczeniami.

* * *

Polskie litery

Szanowna Redakcjo!

Czytając styczniowy numer "Komputera" bardzo mnie zainteresował list profesora Iwo Białynickiego-Biruli dotyczący zastąpienia liter ą,ę,ć,ś itp. kombinacją odpowiedniej litery i znaku "tylda". Uważam, że pomysł jest bardzo prosty i

łatwy do przyswojenia. Jednak mam pewne zastrzeżenia.

Praca przy monitorze - zgoda, ale wydruk na papierze wyglądałby nieelegancko. Dlatego mam propozycję dla programistów, aby zajęli się tym problemem:

1. Napisać program, który przed wydrukiem na papierze przerobi kombinację litery i znaku "tylda" na odpowiednik polski i dopiero przystąpi do druku.

2. Napisać program, który przerobi tę kombinację na polski odpowiednik, ukaże go na monitorze, a dopiero po zaakceptowaniu treści przystąpi do druku.

Mam jeszcze jedną propozycję dotyczącą pisowni polskiej. Dotychczas polskie litery można było uzyskać przez kombinację klawiszy "Shift" lub "Ctrl" i jakiegoś innego. Jest to dobry pomysł, jednak pozbawia nas wielu tak potrzebnych znaków. Mój pomysł polega na tym, że przy pisaniu polskich liter wykorzystywać będziemy tylko jeden klawisz (rzadko używany). W tym przypadku konieczny byłby program, który reagowałby na naciśnięcie tego klawisza, np.:

- sprawdź czy naciśnięty (ż.u.) - tak, idź dalej / nie, powrót;
- zapamiętaj współrzędne kursora;
- zapamiętaj literę, współrzędne-1;
- porównaj z polskim odpowiednikiem;
- wpisz polską literę, współrzędne-1;
- odzyskaj współrzędne kursora;
- wpisz pozycje kursora;
- powrót do programu.

Z pozdrowieniami Wasz stały czytelnik

Krzysztof Petrykowski
Gdańsk

Profesor Białynicki-Birula zgłosił bardzo ciekawą propozycję uwzględniania polskich liter, zakładającą jednakże niewprowadzanie żadnych zmian tak w sprzęcie jak i użytkowanych programach. Wydaje się nam, że pomysł Czytelnika jest tamtej propozycji dobrym uzupełnieniem. Dla wytrawnych programistów zrealizowanie pomysłu nie będzie stanowić większego problemu.

* * *

"Komputer" na cenzurowanym - propozycje

Szanowna Redakcjo!

Czytam Wasze pismo od pierwszego numeru, a każdy numer prawie od deski do deski. Jako stały czytelnik chcę przekazać Redakcji wyrazy uznania za dotychczasową pracę.

Ostatnio jesteśmy świadkami żywej dyskusji na temat treści miesięcznika "Komputer" - są czytelnicy, dla których tematyka "Komputera" jest zbyt łatwa, są i tacy, którzy czytają tylko niektóre artykuły, bo reszta jest zbyt trudna. Zdaję sobie sprawę z tego, że niełatwo jest zaspokoić gusta szerokiego grona czytelników. Moim zdaniem miesięcznik jest redagowany prawie bezbłędnie. Uważam, że zakres tematyczny "Komputera" jest na właściwym poziomie (dla komputerowych przedszkolaków jest "Bajtek", a dla fachowców "Informatyka", no i może "Mikroklan"). "Komputer" bardzo sprawnie wypełnia lukę pomiędzy nimi.

Jeśli mogę coś poradzić to jedynie to, aby Redakcja, mimo krytyki z jednej i z drugiej strony, trzymała się obranego kursu (jeśli chodzi o stopień zaawansowania). Wydaje mi się, że w "Komputerze" powinno znaleźć się trochę więcej materiałów dotyczących procesu tworzenia oprogramowania.

Myślę o przedstawieniu algorytmów rozwiązań różnych problemów. Tu właśnie chcę zwrócić uwagę na to, aby były to algorytmy, a nie gotowe programy (oczywiście czasami specyficzny algorytm trzeba poprzeć konkretnym programem). Może czytelnicy przedstawiliby rozwiązania swoich problemów.

I jeszcze jedno - nie zapominajcie o użytkownikach domowych komputerów ośmiobitowych, których w Polsce jest dość dużo.

Pragnę także wysunąć propozycję stworzenia "Banku literatury komputerowej (informatycznej)". Może istnieje taka możliwość, aby niewielki ułamek jednej ze stron "Komputera" poświęcić dla czytelników, którzy chcą nabyć, odkupić lub wymienić literaturę informatyczną.

Pozostaje mi jeszcze życzyć Redakcji dalszej, tak owocnej jak do tej pory, działalności, której wynikiem jest najlepsze w Polsce (takie jest moje zdanie) pismo komputerowe.

Z poważaniem

Wojciech Kolarz
Chorzów

* * *

Szanowna Redakcjo!

Od dawna nosiłem się z zamiarem napisania do Was. Po przeczytaniu ostatniego numeru "Komputera" (2/88) nie wytrzymałem i napisałem ten list.

Moim i wielu moich kolegów zdaniem ankieta przeprowadzona przez Was nie jest rzeczywistym odbiciem sytuacji na rynku komputerowym w Polsce. Uważam też, że zmiana przeprowadzona przez Was w czasopiśmie (ilość artykułów o Atari) jest bardzo złym rozwiązaniem i w pełni udowadnia Wasz "rasizm" komputerowy (jesteście zdecydowanie po stronie Atari). Większość artykułów o Atari ST robi z niego świętość, nie ukazując żadnych wad tego komputera, który w porównaniu z np. Commodore Amiga jest sprzętem o ubogich możliwościach. Moim zdaniem powinniście utworzyć osobne rubryki dla komputerów u nas popularnych (takich jak Spectrum, Commodore czy Atari). Każda z rubryk zajmowałaby tyle samo miejsca. Wtedy wszystko byłoby już w porządku.

Z poważaniem

Grzegorz Kędzia
Katowice

Krótkie recenzje oraz omówienia najnowszych pozycji literatury komputerowej, jakie ukazały się na naszym rynku, prezentujemy co miesiąc w rubryce "Czytaj!". Jeśli chodzi o rasizm komputerowy, to odpowiadaliśmy na ten nieprawdziwy zarzut wielokrotnie. Propozycja rozbicia pisma na klany nie wydaje się nam ani słuszna, ani uzasadniona.

* * *

Sprawdzajcie dostępne Wam komputery!

Szanowni Państwo!

Informuję, że skradziono mi mikrokomputer Atari 800 XL nr AT 8452029 B-394 wraz ze stacją dysków Atari 1050 nr 92620 454 oraz Joystick Atari CX40, który wewnątrz został wyposażony w mikrowyłączniki.

Komputer ma obok gniazd joysticków dodatkowe gniazdo (żeńskie) 25-otworowe typu Centronix do drukarki. Zasilacz do komputera ma w obudowie powiercone otwory dla chłodzenia - był rozbierany i wewnątrz ma nowy transformator. Zasilacz oryginalny był zalany żywicą. Stacja dysków bez przeróbek. Mam wielką prośbę - sprawdzajcie dostępne Wam komputery! Komputer łatwo rozpoznać na giełdzie lub przy naprawie po dodatkowym złączu.

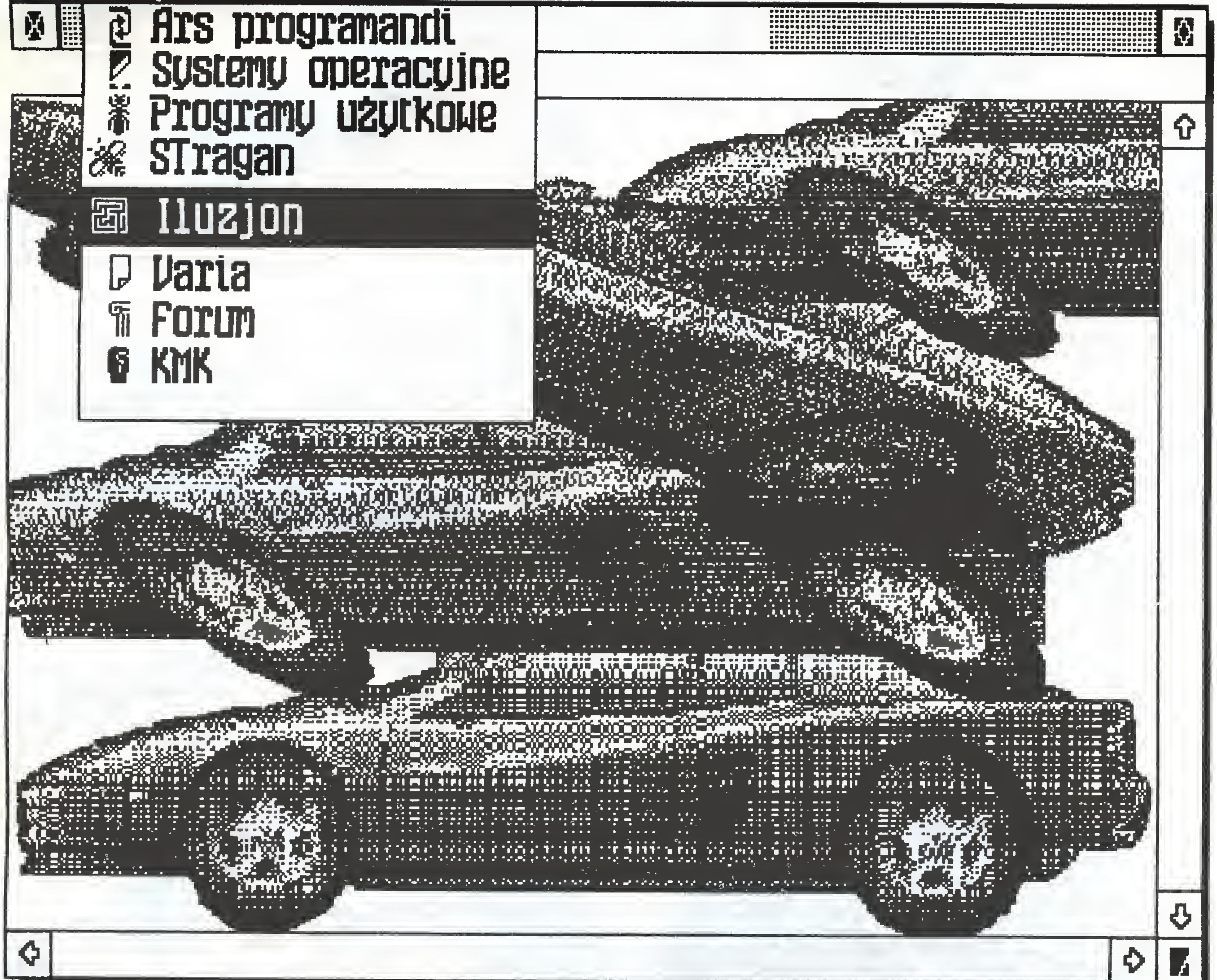
Za pomoc w odzyskaniu komputera i stacji dysków wysoka nagroda - w bonach PKO.

Zgłoszenia proszę kierować:

Jerzy Kaleta
ul.Czarnowiejska 75/85 m. 17
30-049 Kraków

Drukując list mamy nadzieję, że pomożemy naszemu Czytelnikowi.





\Wdomu\

Mariusz Pietruszka, Tadeusz Jedynak

System operacyjny CP/M - 80

Autorem założeń i pierwszych wersji systemu CP/M jest Gary Kidall. Pierwsza wersja użytkowa powstała w połowie lat siedemdziesiątych. Rozwijana w kolejnych latach doczekała się wersji 3.0 dla komputerów 8-bitowych oraz wersji CP/M 86 - dla komputerów PC.

Obecnie CP/M i jego znak firmowy są własnością firmy Digital Research, Inc.

CP/M 3.0 jest popularnym systemem operacyjnym dla mikrokomputerów 8-bitowych. Jest to system uniwersalny, jednoterminalowy (choć istnieje jego wielodostępna mutacja - MP/M). Jako system operacyjny CP/M zarządza pamięcią, konsolą operatorską (ekran + klawiatura), drukarką oraz innymi urządzeniami komunikacyjnymi wejścia-wyjścia. CP/M zarządza również informacją zgromadzoną w zbiorach dyskowych. Może np. skopiować zbiór z dysku do pamięci komputera lub do urządzeń peryferyjnych takich jak drukarka. Aby to wykonać, CP/M umieszcza różne programy w pamięci i wykonuje je w odpowiedzi na rozkazy wprowadzane z konsoli. Programy te wykonują się jako sekwencja kroków.

CP/M-u można używać do stworzenia własnych programów lub skorzystać z bogactwa dostępnych programów aplikacyjnych.

Minimalne wymagania sprzętowe dla CP/M 3.0 to: komputer zawierający procesor 8080, 8085, Z80 lub inny równoważny (np. HD64180), urządzenie konsoli (ogólnie: klawiatura i wyświetlacz, np. ekran monitora (CRT)) oraz przynajmniej jeden sterownik dysków elastycznych. Aby móc wykorzystać wszystkie możliwości CP/M-u, nieodzowne są dwie stacje dyskietek.

System ten jest logicznie podzielony na cztery części:

BIOS (ang. *Basic Input/Output System*) - zestaw procedur obsługi podstawowych urządzeń zewnętrznych. Za wyjątkiem modułu Kernal BIOS (jądro) istotnie różni się w różnych komputerach pracujących pod nadzorem CP/M.

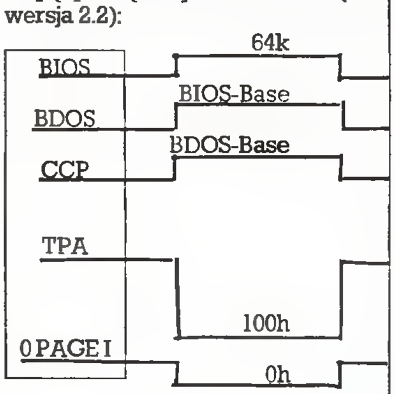
BDOS (ang. *Basic Disk Operating System*) - zestaw procedur obsługi pamięci dyskowej i zarządzania plikami. Jest taki sam dla danej wersji systemu i programowo zgodny z poprzednimi wersjami (niezależnie od sprzętu).

CCP (ang. *Console Command Processor*) - zespół procedur rozpoznających i realizujących zlecenia użytkownika systemu.

TPA (ang. *Transient Program*

Area) - jest wolnym obszarem pamięci operacyjnej (RAM) przeznaczonym dla programów użytkowych. Obszar ten zaczyna się od adresu 100h.

Możemy zatem stworzyć prostą mapę pamięci systemu CP/M (tu: wersja 2.2):



Po włączeniu komputera i zainstalowaniu systemu operacyjnego CP/M (co sprowadza się do włożenia dyskietki systemowej, z której system ten zostaje przepisany do pamięci komputera) zgłasza on swą gotowość do przyjmowania zleceń poprzez wyświetlenie na konsoli tzw. zgłoszenia systemowego (ang. *system prompt*, w luźnym tłumaczeniu: znak zachęty): **A>**

System jest gotowy do przyjmowania zleceń z klawiatury. W tym momencie OPERATOR może wpisywać zlecenia dla systemu operacyjnego. Po naciśnięciu klawisza **RETURN** (lub **ENTER**) system interpretuje i wykonuje otrzymany rozkaz. W przypadku gdy rozkaz jest nazwą programu, zostaje on przepisany z dyskietki do pamięci głównej i wykonany.
Wpisanie **A>DIR** <cr>
spowoduje wypisanie katalogu zbiorów dyskowych na ekranie monitora. Poprzez **A>MBASIC** <cr> uruchamiamy interpreter Basica i znajdujemy się w dobrze znanej nam sytuacji PROGRAMISTY z C-64 lub ZX Spectrum. Możemy wówczas napisać linię:
10 PRINT "Jestem w Basicu"
<cr>
i uruchomić program poprzez zwykłą instrukcję RUN. Po wpisaniu instrukcji **SYSTEM** nastąpi powrót do systemu operacyjnego.

Startowanie lub ładowanie systemu operacyjnego oznacza czytanie kopii systemu operacyjnego z dyskietki systemowej do pamięci komputera. Szczegóły tej operacji będą różne w różnych komputerach - odsyłamy do instrukcji obsługi. Po załadowaniu zostanie wypisana informacja o następującej (lub podobnej) treści:

CP/M 3.00 Version V.V
informująca o posiadanej wersji systemu. Następnie pojawi się wspomniane we wstępie zgłoszenie systemowe: **A>** . Znak "A" oznacza przypisany napęd. Oznacza to, że dopóki OPERATOR nie zadecyduje inaczej, CP/M będzie szukał programów i zbiorów danych na dyskietce w napędzie A. Zgłoszenie to informuje nas również, że aktualnym użytkownikiem jest użytkownik o numerze 0.

Linia rozkazowa CP/M składa się z rozkazu kluczowego i opcjonalnego rozszerzenia oraz "powrotu karetki" (naciśnięcie klawisza RETURN). Słowo kluczowe identyfikuje rozkaz lub program do wykonania. Naciśnięcie klawisza RETURN lub ENTER będziemy oznaczali: <cr>.

CP/M 3 rozpoznaje dwa typy rozkazów: rezydentne i nierezydentne (ang. *transient* - przemijające). Rozkazy rezydentne wykonują programy, które znajdują się w pamięci jako część systemu operacyjnego CP/M 3. Mogą być zatem wykonywane natychmiast. Rozkazy nierezydentne są zgromadzone na dysku jako programy. Muszą być zatem załadowane najpierw do pamięci, aby wypełnić swe zadanie. Rozkazy nierezydentne można obejrzeć w katalogu dyskietki systemowej, gdyż ich nazwy związane są z rozszerzeniem **COM**, np. **PIP.COM.** Rozkazy nierezydentne po przepisaniu z dyskietki zajmują obszar pamięci operacyjnej **TPA** od adresu 100h.

Pliki dyskowe

Najważniejszym zadaniem CP/M-a jest obsługa plików dyskowych. CP/M może tworzyć (ang. *create*), czytać (*read*), zapisywać (*write*), kopiować (*copy*) lub wymazywać (erase) pliki dyskowe.

Czym jest plik?

Plik pod kontrolą systemu CP/M jest zbiorem związanych wzajemnie informacji magazynowanych na dysku. Każdy plik musi mieć jednoznaczną nazwę, ponieważ system używa tej nazwy, aby uzyskać do niego dostęp. Katalog dyskowy (ang. *directory*) jest przechowywany również na dysku. Katalog ten zawiera informacje o zbiorach (plikach) dyskowych i o lokalizacji tych zbiorów na dyskietce.

W CP/M mamy do dyspozycji dwa rodzaje plików: programowe (ang. *program files*) i pliki danych (ang. *data files*). Plik programowy zawiera program, który jest ciągiem instrukcji wykonywanych przez komputer krok po kroku. Plik danych jest zazwyczaj pewnym zbiorem informacji: np. listą nazwisk i adresów, spisem magazynowym, tekstem dokumentu lub podobnie powiązanymi informacjami. Komputer, rzecz jasna, nie może wykonywać nazwisk i adresów, może natomiast wykonać program, który będzie je drukował na kopertach. Plik danych może również zawierać program źródłowy, lecz musi być on przetworzony w programach typu asembler lub kompilator zanim stanie się plikiem programowym. W większości przypadków działający program przetwarza pliki danych (mogą to być np. ciągi liczb otrzymane w wyniku doświadczenia). Oczywiście zdarza się czasami sytuacja, kiedy wykonywany program przetwarza inny plik programowy, np. program PIP.COM może kopiować jeden lub kilka innych plików programowych.

Jak utworzyć plik?

Jest wiele sposobów tworzenia plików. Jednym z nich jest użycie edytora tekstów. Liniowy edytor ED.COM pozwala nam na utworzenie zbioru tekstowego i nadanie mu nazwy. Innym sposobem jest tworzenie nowych plików poprzez np. kopiowanie zbiorów wraz ze zmianą nazwy.

Jak utworzyć nazwę pliku?

CP/M 3 identyfikuje każdy plik poprzez jednoznaczne określenie jego nazwy. Plik jest określony przez nazwę składającą się z 1 do 8 znaków, np. MOJPLIK. Plik może być w pełni określony przez: przypisanie logicznego napędu dyskowego, nazwę, typ oraz hasło z nim związane.

Napęd dyskowy jest określony przez pojedynczą literę (od A do P), po której następuje dwukropek. Jeśli zatem wprowadzimy: **B: MOJPLIK**
CP/M 3 będzie szukał tego zbioru w napędzie B. Może być to ten sam napęd fizyczny - np. CP/M w Amstradzie CPC 6128 najpierw zażąda zmiany dyskietki. Doświadczenie uczy, że nazwa powinna coś oznaczać dla użytkownika tworzącego dany plik. Jeśli zatem będziemy chcieli utworzyć plik danych z nazwiskami osób posiadających komputery pracujące pod nadzorem tego samego systemu operacyjnego, możemy go np. nazwać **KOMPUNAZ.CPM**, gdzie występujące po kropce rozszerzenie nazwy wskazuje nam na pewną kategorię obiektów. Zbiór nazwisk użytkowników MS-DOS-a tworzyłby tu **KOMPUNAZ.DOS**, etc. Typ, nazywany często rozszerzeniem nazwy zbioru, niekoniecznie musi być określony, jeśli jednak występuje, nie może mieć więcej niż 3 znaki i musi występować po kropce, jak pokazano w powyższym przykładzie. W wersji tzw. bankowanej CP/M 3 (ang. *banked*) można dodać hasło jako opcjonalną część specyfikacji zbioru. Hasło dołącza się do nazwy pliku po średniku. Nie może mieć ono więcej niż 8 znaków. W naszym przykładzie: **KOMPUNAZ.CPM;TAJNE**
Jeśli zbiór jest chroniony przez hasło, konieczne jest wpisanie hasła, aby uzyskać dostęp do niego.

W tworzeniu nazw, typów i haseł nie wolno używać znaków, które mają specjalne znaczenie dla CP/M:

< > = , ! * ? & / $ [ ] ( ) . : ; + -

Nasz przykład z wyszczególnieniem wszystkich możliwych elementów opisu pliku będzie wyglądał następująco:
A:KOMPUNAZ.CPM; TAJNE

A skoro tajne, to trzeba poczekać na zezwolenie dostępu do zbiorów. Trwa to około miesiąca...

W domu

Cezary Waśniewski

Algorytmy sortowania

Każdy początkujący programista zetknie się prędzej czy później z problemem uporządkowania pewnego zbioru elementów. Wielu młodych entuzjastów programowania zechce zmierzyć się z nimi samodzielnie, większość jednak wolałaby skorzystać z istniejących już rozwiązań. Oto kilka algorytmów sortowania wraz z krótkimi opisami, mającymi za zadanie pomóc w samodzielnej ich analizie.

Załóżmy na wstępie, że do uporządkowania mamy tablicę tekstową A$, składającą się z X elementów o długości Y znaków, tzn. deklarowaną instrukcją DIM A$(X,Y). (Algorytmy zapisano w języku Basic dla komputera ZX Spectrum.)

Zaczniemy od najprostszego chyba algorytmu, zwanego wybieraniem prostym (ang. *straight select*), przedstawionego na listingu nr 1.

```
   1 REM ****LISTING NR 1****
   2 REM
   3 REM WYBIERANIE PROSTE
   4 REM
  10 FOR N=X TO 2 STEP -1
  20 LET P$=""
  30 FOR M=1 TO N
  40 IF A$(M)>P$ THEN LET P$=A$(
M): LET P=M
  50 NEXT M
  60 LET T$=A$(N): LET A$(N)=A$(
P): LET A$(P)=T$
  70 NEXT N
```

Rozpoczynamy tutaj od przeglądu wszystkich elementów tablicy A$, wyszukujemy największy jej element, zapamiętujemy go, zamieniamy miejscami z ostatnim elementem tablicy. W następnym kroku przeszukujemy w ten sam sposób tablicę A$ (oprócz ostatniego pola) i największy z elementów odkładamy na pole przedostatnie. Wyczerpanie pętli 10-70 prowadzi do całkowitego uporządkowania tablicy A$.

Przejdźmy teraz do następnego z algorytmów prostych, zwanego wstawianiem prostym (ang. *straight insertion*) przedstawionego na listingu nr 2.

```
   1 REM ****LISTING NR 2****
   2 REM
   3 REM   WSTAWIANIE PROSTE
   4 REM
  10 FOR N=2 TO X
  20 IF A$(N)>=A$(N-1) THEN GO T
O 90
  30 LET M$=A$(N)
  40 FOR M=N-1 TO 1 STEP -1
  50 LET A$(M+1)=A$(M)
```

> 18

17 <

```
55 IF M=1 THEN GO TO 70
60 IF M$<=A$(M-1) THEN GO TO 8
0
70 LET A$(M)=M$: GO TO 90
80 NEXT M
90 NEXT N
```

Sortowanie rozpoczynamy od porównania dwóch początkowych elementów tablicy A$. Jeśli ich porządek jest nieprawidłowy, są one zamieniane miejscami.

Następnie element trzeci jest porównywany z elementem drugim, a w przypadku nieprawidłowego porządku z elementem pierwszym oraz wstawiany we właściwe miejsce tej części tablicy A$, przy czym elementy większe są przesuwane o jedno miejsce w kierunku pól o wyższych numerach. Postępując dalej w ten sam sposób, aż do elementu ostatniego, uzyskamy całkowite uporządkowanie tablicy A$.

Następny z przedstawianych tu algorytmów prostych - sortowanie bąbelkowe (ang. *bubblesort*) - jest stosunkowo dobrze znany czytelnikom polskich pism informatycznych. Przedstawiony został na listingu nr 3.

```
 1 REM ****LISTING NR 3****
 2 REM
 3 REM SORTOWANIE BABELKOWE
 4 REM
10 FOR N=X-1 TO 1 STEP -1
20 FOR M=1 TO N
30 IF A$(M)<=A$(M+1) THEN GO T
0 50
40 LET P$=A$(M): LET A$(M)=A$(
M+1): LET A$(M+1)=P$
50 NEXT M
60 NEXT N
```

Sortowanie bąbelkowe polega na przeglądaniu sortowanej tablicy A$ od pola pierwszego w kierunku pól wyższych, z kolejnym porównywaniem par elementów sąsiednich i zamianie tych elementów miejscami w przypadku nieprawidłowego uporządkowania (w parze).

Przedstawiony tu algorytm ma widoczną na pierwszy rzut oka, istotną wadę. Przebiega on mianowicie niezależnie od wyjściowego uporządkowania tablicy A$ (o ile istnieje). Kolejne bowiem przejście przez tablicę A$ zachodzi niezależnie od tego, czy w przejściu poprzednim dokonane były jakieś przestawienia, czy nie. Tę wadę można w prosty sposób usunąć. Poprawiony algorytm sortowania bąbelkowego przedstawiony został na listingu nr 4.

```
  1 REM ****LISTING NR 4****
  2 REM
  3 REM SORTOWANIE BABELKOWE
          ULEPSZONE
  4 REM
 10 LET P=X-1
 20 FOR N=X-1 TO 1 STEP -1
 30 LET F=0
 40 FOR M=1 TO P
 50 IF A$(M)<=A$(M+1) THEN GO T
0 70
 60 LET F=M: LET P$=A$(M): LET
A$(M)=A$(M+1): LET A$(M+1)=P$
 70 NEXT M
 80 LET P=F: IF F=0 THEN GO TO
100
 90 NEXT N
100 REM KONIEC
```

Łatwo zauważyć dwie różnice w stosunku do algorytmu poprzedniego. Sprawdza się tu, czy w danym przejściu nastąpiło jakieś przestawienie elementów. Jeśli nie, tzn. ukończono sortowanie, następuje przejście do końca algorytmu. Ponadto jest zapamiętywana pozycja ostatnio przeprowadzonego przestawienia elementów i w następnym przejściu przeszukiwanie dochodzi już tylko do tego miejsca. Widać, że tak poprawiony algorytm nie będzie istotnie szybszy w przypadku tablicy A$, w której elementy ułożone są zupełnie przypadkowo. Daje on jednak olbrzymie zyski czasowe w przypadku sortowania tablic o znacznym stopniu uporządkowania, na przykład takich, w których po uprzednim uporządkowaniu wprowadzono stosunkowo niewiele zmian.

Przejdziemy teraz do bardziej złożonych algorytmów sortowania. Zajmiemy się najpierw algorytmem opisanym po raz pierwszy przez D. L. Shella w roku 1959 *(Shellsort)*, przedstawionym na listingu nr 5.

```
  1 REM ****LISTING NR 5****
  2 REM
  3 REM       SHELLSORT
  4 REM
 10 DIM B$(X,Y)
 20 LET P=INT (X/2)
 30 FOR N=1 TO P
 40 FOR M=1 TO INT (X/P)
 50 LET B$(M)=A$((M-1)*P+N)
 60 NEXT M
 70 LET XX=M-1: GO SUB 200
 80 FOR M=1 TO INT (X/P)
 90 LET A$((M-1)*P+N)=B$(M)
100 NEXT M
110 NEXT N
120 LET P=INT (P/2)
130 IF P<>0 THEN GO TO 30
140 GO TO 300
150 REM WSTAWIANIE PROSTE
200 FOR K=2 TO XX
210 IF B$(K)>=B$(K-1) THEN GO T
O 270
220 LET P$=B$(K): FOR L=K-1 TO
1 STEP -1
230 LET B$(L+1)=B$(L)
240 IF P$<=B$(L-1) THEN GO TO 2
60
250 LET B$(L)=P$: GO TO 270
260 NEXT L
270 NEXT K
280 RETURN
300 REM KONIEC
```

Działanie algorytmu *Shellsort* najłatwiej zrozumieć na przykładzie niewielkiej (np. 10-elementowej) tablicy A$. Na początku zmiennej P przypisuje się wartość połowy wymiaru tablicy A$ (P=5, linia 20). Następnie brane są pod uwagę pary: 1 i 6 element, 2 i 7 element, ..., 5 i 10 element, traktowane jako oddzielne tablice pomocnicze B$ i porządkowane za pomocą wstawiania prostego (linie 200-280). Po uporządkowaniu elementy par wracają do tablicy A$. Następnie powtórnie dzielimy na połowę wartość zmiennej P, otrzymując P=2. Tym razem bierzemy pod uwagę dwie pomocnicze tablice B$ o wymiarze 5, złożone z elementów: pierwsza - 1., 3., 5., 7. i 9. tablicy A$, druga - 2., 4., 6., 8. i 10. tablicy A$. Te dwie tablice są sortowane oddzielnie i ich elementy wracają na właściwe miejsca tablicy A$. Ostatnim posunięciem (P=1) jest ostateczne uporządkowanie tablicy A$ za pomocą procedury 200-280.

Mniej uważny czytelnik mógłby zapytać, po co w sposób sztuczny wydzielać i sortować części tablicy A$, by w końcu sortować ją w całości. Czytelnik, który zechce uważniej prześledzić, a może wypróbować działanie algorytmu na konkretnej (niewielkiej) tablicy, zauważy, że po kolejnych posunięciach tablica A$ jest coraz bardziej uporządkowana, co powinno istotnie wpływać na szybkość wykonywania algorytmu.

Zwróćmy uwagę, że procedura pomocnicza może wykorzystywać dowolny z przedstawionych wcześniej prostych algorytmów sortowania. Najlepiej wypróbować wszystkie i wybrać taki, który zapewnia najszybsze działanie całego algorytmu *Shellsort*.

Na zakończenie zajmiemy się algorytmem znanym pod nazwą *Quicksort*, po raz pierwszy opisanym przez R. Hoare w 1962 r. Algorytm ten godny jest szczególnej uwagi, gdyż łączy w sobie niebywałą prostotę z wielką efektywnością działania. *Quicksort* przedstawiono na listingu nr 6.

```
  1 REM ****LISTING NR 6****
  2 REM
  3 REM       QUICKSORT
  4 REM
 10 DIM L(X): DIM P(X). LET T=0
 LET L(1)=1: LET P(1)=X
 20 GO SUB 40
 30 GO TO 200
 40 LET T=T+I: IF L(T)>=P(T) TH
EN GO TO 150
 50 LET A=L(T): LET B=P(T)
 60 LET W$=A$(INT ((A+B)/2))
 70 IF A>B THEN GO TO 130
 80 IF A$(A)<W$ THEN LET A=A+1:
 GO TO 80
 90 IF A$(B)>W$ THEN LET B=B-1:
 GO TO 90
100 IF A>B THEN GO TO 130
110 LET T$=A$(A): LET A$(A)=A$(
B): LET A$(B)=T$
120 LET A=A+1: LET B=B-1: GO TO
 70
130 LET P(T+1)=B: LET L(T+1)=L(
T): GO SUB 40
140 LET L(T+1)=A: LET P(T+1)=P(
T): GO SUB 40
150 LET T=T-1: RETURN
200 REM KONIEC
```

Zasada działania tego algorytmu jest niezwykle prosta. Z tablicy A$ wybieramy dowolny element i jego wartość zapisujemy jako wartość pomocniczej zmiennej łańcuchowej W$. Służy ona dalej jako wartość porównawcza dla pozostałych elementów tablicy A$. Następnie wszystkie elementy mniejsze od wybranego elementu W$ przekładane są w tablicy A$ na pozycje niższe, a elementy większe na pozycje wyższe od wybranej. Z dwu części tej trochę już uporządkowanej tablicy A$ wybieramy dowolne elementy jako wartości porównawcze i przeprowadzamy operacje (podobnie jak w pierwszym podziale) oddzielnie dla każdej części. Postępując dalej w ten sam sposób dojdziemy do momentu, gdy części tablicy A$ podlegające kolejnym podziałom będą zawierały po jednym elemencie, a zatem sortowanie tablicy A$ zostanie zakończone.

Należy zwrócić uwagę na fakt, że program *Quicksort* jest prostym przykładem procedury rekursywnej tzn. odwołującej się do samej siebie.

Przegląd algorytmów sortowania

Przy ocenie algorytmów sortowania brano pod uwagę dwa główne czynniki: szybkość działania i pamięć zajmowaną przez sam program sortujący.

Algorytm sortujący	Czas sortowania [s] A$(100,3)	A$(500,3)
1. Wybieranie proste	64	1446
2. Wstawianie proste	71	1780
3. Sortowanie bąbelkowe	133	3396
4. Sortowanie bąbelkowe ul.	143	3722
5. SHELLSORT	57	383
6. QUICKSORT	38	222

Rys. 1. Czasy sortowania tablic

Badając przedstawione powyżej programy 1 - 6 na komputerze ZX Spectrum, zmierzono czasy sortowania 100 i 500 ustawionych przypadkowo słów trzyliterowych. Wyniki pomiarów przedstawiono na rys. 1.

Jak widać, porównanie wypada zdecydowanie na korzyść bardziej rozbudowanych algorytmów, zwłaszcza w miarę wzrostu wymiaru sortowanych tablic. Następny wniosek: tak chętnie opisywany algorytm sortowania bąbelkowego w przypadku tablic o przypadkowo rozłożonych elementach nie wytrzymuje porównania ze znacznie zresztą prostszymi algorytmami wybierania i wstawiania. Rozważając drugi składnik efektywności algorytmów sortowania dochodzimy do wniosku, że szybkość działania musi być okupiona zwiększeniem objętości pamięci operacyjnej, wykorzystywanej przez sam algorytm.

W przypadku algorytmów prostych nie ma w zasadzie problemu: wykorzystują najwyżej kilka zmiennych pomocniczych do przechowywania wartości i sterowania pętlami. Zmienne te bez kłopotu wykorzystać można w innych częściach programu. Dużo gorzej dzieje się w przypadku bardziej skomplikowanych algorytmów. Algorytm Shella wymaga zadeklarowania pomocniczej tablicy B$ o wymiarze takim samym, jak sortowana tablica A$. Zmniejsza to użyteczność algorytmu w przypadku większej bazy danych. Przypomnę, że np. w ZX Spectrum raz zgłoszona deklaracja tablicy rezerwuje pamięć dla odpowiedniej liczby elementów i nawet gdy nie będziemy już w programie korzystać z pomocniczej tablicy B$, nie możemy zwolnić zarezerwowanego dla niej pola pamięci. Tutaj ujawnia się następna zaleta algorytmu *Quicksort*. Wymaga on tylko zadeklarowania dwóch tablic liczbowych o wymiarze takim samym, jak sortowana tablica A$. W praktycznych zastosowaniach (dość długie teksty składowane w tablicy) daje to znaczną oszczędność pamięci w porównaniu z algorytmem *Shellsort*.

Ps. O sortowaniu pisaliśmy już w "Komputerze" w numerach: 4/86, 7/86, 5/87.

W domu

Przemysław Skiba

Amstrad CPC i okna

Dla wygody użytkownika w wielu programach stosuje się różnorodne ułatwienia w postaci ściągawek, rozwijanych menu, a nawet całych ekranów z instrukcją obsługi. W zastosowaniach nie stawiających wysokich wymagań stronie graficznej jest to zadanie dość proste do zrealizowania.

Nieco inaczej jest w przypadkach, w których grafika stanowi istotną część programu - wyświetlenie dodatkowych opisów pociąga za sobą skasowanie zawartości ekranu znajdującej się w tym miejscu. Pół biedy, jeśli zmazany fragment zawierał prostą, łatwą do ponownego skonstruowania grafikę. Problemy zaczynają się, gdy zawartość ekranu jest na tyle złożona, że jej każdorazowe odtwarzanie byłoby bardzo czasochłonne.

Proponuję krótką procedurę (50 bajtów) napisaną w języku asemblera komputerów serii Amstrad CPC, umożliwiającą wymianę zawartości dowolnego prostokątnego fragmentu ekranu (okna) i zdefiniowanego uprzednio bufora w każdym trybie graficznym. Najmniejszym możliwym do zapamiętania i odtworzenia obszarem jest zawartość jednego bajta ekranu, największym - cały ekran. Za pomocą procedury można z poziomu Basica tworzyć dowolną ilość okien. Jedynym ograniczeniem jest dostępna pamięć.

Okna mogą być pasywne - odtwarzające wcześniej przygotowaną zawartość bufora (np. menu), lub aktywne - pozwalające na zmianę swojej zawartości podczas pracy programu (np. notes).

Procedura została opracowana w formie relokowalnej, aby umożliwić Czytelnikom wykorzystanie jej we własnych programach. Wszystkie parametry przekazywane są z Basica.

Rys. 1 przedstawia tekst procedury w wersji dla asemblera GENA3 firmy HiSOFT.

```
100 ;****************************************
110 ;*      SWAPPER    (c)1987 P.SKIBA      *
120 ;* CALL ORG,ekran,bufor,linie,szerokosc *
130 ;*        procedura relokowalna         *
140 ;****************************************
150
160          ORG   41500              ; adres SWAPPER'a
170 nextln:  EQU   #BC26              ; procedura ROM'u
180 ;-------------------------------
190 ;        Pobranie zmiennych      ;
200 ;-------------------------------
210          ld    H,(IX+7)           ; adr. pierwszego
220          ld    L,(IX+6)           ; bajtu ekranu
230          ld    D,(IX+5)           ; adr. pierwszego
240          ld    E,(IX+4)           ; bajtu bufora
250          ld    B,(IX+2)           ; ilosc linii
260          ld    C,(IX+0)           ; szerokosc linii
270 ;-------------------------------
280 ;        Wymiana zawartosci      ;
290 ;        ekranu i bufora         ;
300 ;-------------------------------
310 petlaA:  push  BC                 ; zachowaj ilosc linii
320 petlaB:  ld    A,(DE)             ; wez zawartosc bufora
330          ld    B,(HL)             ; wez zawartosc ekranu
340          ld    (HL),A             ; przenies zawartosc
350 ;                                   bufora na ekran
360          ld    A,B
370          ld    (DE),A             ; przenies zawartosc
380 ;                                   ekranu do bufora
390          inc   HL                 ; zwieksz adres ekranu
400          inc   DE                 ; i bufora
410          dec   C                  ; zmniejsz szerokosc
420          ld    A,C
430          cp    0                  ; powtorz jesli nie
440          jr    NZ,petlaB          ; koniec linii
450 ;-------------------------------
460 ;  Obliczanie adresu pierwszego ;
470 ;  bajtu nastepnej linii ekranu ;
480 ;-------------------------------
490          push DE
500          ld   D,0                 ; wez poczatkowa
510          ld   E,(IX+0)            ; szerokosc linii
520          sbc  HL,DE               ; wroc do poczatku linii
530          pop  DE
540          call nextln              ; oblicz adr. nast. linii
550          pop  BC                  ; przywroc ilosc linii
560          djnz petlaA              ; zmniejsz ilosc linii,
570 ;                                   nie wszystkie-powtorz
580          ret                      ; powrot do BASIC'a
```

1

Czytelnikom nie posiadającym asemblera proponuję wprowadzić krótki program demonstracyjny (rys. 2), w skład którego wchodzi procedura ładująca kod maszynowy (linie 160- 290) pod adres wskazany w linii 160. W razie błędnego wpisania danych (linie 270-290) zostanie wyświetlona uwaga z numerem linii, w której został popełniony błąd. Uruchomienie programu będzie możliwe tylko w przypadku poprawnego wprowadzenia danych. Program pokazuje zastosowanie aktywnego okna we wszystkich trybach graficznych ekranu.

2

```
100 '**********************************
110 '* SWAPPER DEMO   (C)1987 P.SKIBA *
120 '**********************************
130 '
140 '*** LADOWANIE KODU MASZYNOWEGO ***
150 '--------------------------------
160 MEMORY 41500:adres=HIMEM+1:SWAPPER=adres
170 FOR i=1 TO 3
180  ok=0:READ kod$,suma$
190  FOR j=1 TO 34 STEP 2
200   kod=VAL("&"+MID$(kod$,j,2))
210   POKE adres,kod
220   ok=ok+kod:adres=adres+1
230  NEXT
240  IF ok=VAL("&"+suma$) THEN 260
250  PRINT"*** BLAD W LINII";260+i*10;"****":END
260 NEXT
270 DATA DD6E06DD6607DD5E04DD5605DD4602DD4E,762
280 DATA 00C51A46777812231 30D79FE0020F3D516,5DE
290 DATA 00DD5E00ED52D1CD26BCC110E3C9C90000.840
300 '
310 '*** INICJACJA ZMIENNYCH / ZAKONCZENIE ***
320 '----------------------------------------
330 MODE 1:PRINT"Mode (0,1,2)  (K)oniec"
340 a$=UPPER$(INKEY$)
350  IF a$="0" THEN b=4:xmax=20
360  IF a$="1" THEN b=2:xmax=40
370  IF a$="2" THEN b=1:xmax=80
380  IF a$="K" THEN CLS:END
390 IF NOT(a$="0" OR a$="1" OR a$="2" OR a$="K") THEN 340
400 MODE VAL(a$):BORDER 4:PAPER 0:PEN 1:CLS
410
420 '*** PRZYGOTOWANIE OKIEN ***
430 '--------------------------
440 x1=2:x2=xmax/4:y1=2:y2=16
450 WINDOW#1,x1,x2,y1,y2:PAPER#1,1:PEN#1,0:CLS#1
460 LOCATE 1,23:PRINT STRING$(xmax,"-")
470 LOCATE 1,24:PRINT"Wpisz tresc okna":LINE INPUT#1,okno$
480 ekran=&C000+(y1-1)*80+(x1-1)*b:bufor=adres
490 linie=(y2-y1+1)*8:szerokosc=(x2-x1+1)*b
500 GOSUB 630
510 WINDOW#2,1,xmax/2,1,20:PAPER#2,0:PEN#2,1:CLS#2
520 LOCATE 1,24:PRINT"Wpisz tresc tla ":LINE INPUT#2,tlo$
530 LOCATE 1,24:PRINT"Okno Zmiana Powtorz"
540 '
550 '*** PETLA STERUJACA ***
560 '---------------------
570 a$=UPPER$(INKEY$)
580  IF a$="O" THEN GOSUB 630:WHILE INKEY$="":WEND:GOSUB 630
590  IF a$="Z" THEN GOSUB 630:CLS#1:LINE INPUT#1,okno$:GOSUB 630
600  IF a$="P" THEN 330
610 GOTO 570
620 '-------
630 CALL SWAPPER,ekran,bufor,linie,szerokosc:RETURN
```

> 20

19 <

Zwolenników programowania strukturalnego powinna zainteresować wersja SWAPPER-a napisana dla kompilatora HiSoft Pascal HPT4 (rys. 3).

```
100 PROGRAM SWAPPERPAS;
110 {---(c) P.Skiba---}
120
130 VAR znak : char;
140        n : integer;
150
160 PROCEDURE mode(i:integer);
170 BEGIN
180   rA:=chr(i);
190   user(#BC0E); {procedura ROM}
200 END;
210
220 PROCEDURE keyread;
230 BEGIN
240   user(#BB18); {procedura ROM}
250   znak:=rA;
260 END;
270
280 PROCEDURE swapper(ekr,buf,lin,szer : integer);
290 VAR i,j : integer;
300     exb : char;
310 BEGIN
320   FOR i:=1 TO lin DO BEGIN
330     FOR j:=1 TO szer DO BEGIN
340       exb:=peek(ekr,char);
350       poke(ekr,peek(buf,char));
360       poke(buf,exb);
370       ekr:=ekr+1;buf:=buf+1;
380     END;
390     ekr:=ekr-szer+(#800);
400     IF ekr>(#FFFF) THEN ekr:=ekr-(#3FB0);
410   END;
420 END;
430
440 BEGIN {demo}
450   mode(2);znak:=chr(0);
460   FOR n:=1 TO 3 DO writeln('Zawartosc bufora!');
470   swapper(#C000,#5B68,24,17);
480   page;
490   FOR n:=1 TO 5 DO writeln('Dowolna zawartosc ekranu')
500   writeln;
510   writeln('Nacisnij [ENTER] lub [K]oniec');
520   WHILE NOT(znak IN ['k','K']) DO BEGIN
530     keyread;
540     swapper(#C000,#5B68,24,17);
550   END;
560 END. {demo}
```

Obsługa procedury
Wywołanie: CALL adres,ekran,bufor,linie,szerokość
gdzie:
adres - adres procedury,
ekran - rzeczywisty adres górnego lewego rogu okna,
bufor - adres bufora,
linie - wysokość okna (cały ekran to 25*8=200 linii),
szerokość - ilość bajtów w każdej linii okna.

Pamięć ekranu Amstrada rozpoczyna się od adresu &C000 i składa się z 8 bloków (po 2048 bajtów) opisujących jedną z 8 linii każdego wiersza. Ilość znaków w wierszu zmienia się w zależności od trybu pracy. Różnice w sposobie organizacji ekranu przedstawia poniższa tabela:

tryb	ilość znaków	szerokość znaku
MODE 2	80	1 bajt
MODE 1	40	2 bajty
MODE 0	20	4 bajty

Pamiętając, że szerokość ekranu jest stała i wynosi 80 bajtów dla okna o wymiarach x1,x2,y1,y2 (patrz instr. Window w Basicu), poszczególne parametry można otrzymać stosując wzory:

ekran=&C000+(y1-1)*80+(x1-1)*b (b - szerokość znaku)
linie=(y2-y1+1)*8
szerokość=(x2-x1+1)*b
rozmiar=linie*szerokość (liczba bajtów w oknie)

W przypadku pracy z kilku oknami jednocześnie adres n- tego bufora obliczamy (po przypisaniu zmiennej bufor (0) początkowej wartości obszaru buforów) w następujący sposób:
bufor (n)=bufor (n-1)+rozmiar (n-1)

W domu

Sławomir Zawisza

Atari-writer Plus

Nadchodzi być może już era, w której atrybutem człowieka posługującego się "piórem" stanie się zestaw komputerowy. Ciekawe tylko czy "dzieła komputerowe" przyszłych twórców będą równie skrzętnie przechowywane w muzeach, jak dzisiaj rękopisy najznamienitszych z lat minionych?! Edycja tekstu należy bowiem do najbardziej popularnych profesjonalnych zastosowań mikrokomputerów. Programy tego typu umożliwiają przygotowanie, korektę oraz nadanie właściwej formy tekstom o różnym przeznaczeniu. Użytkownicy Atari dysponują kilkoma edytorami tekstu, spośród których na uwagę zasługuje Atariwriter Plus.

Jest to program umożliwiający zaawansowane redagowanie tekstu. Oprócz operacji na fragmentach tekstu pozwala on na dosyć swobodne kształtowanie jego formy i stylu przed wydrukowaniem. Daje możliwość automatycznego poprawiania błędów dzięki programowi korektorskiemu (Proofreader), posiada też moduł obsługi korespondencji. Atariwriter Plus wykorzystuje dodatkowe bloki pamięci operacyjnej w modelu 130 XE (przełączanie banków następuje przez przyciśnięcie klawiszy START + B) oraz współpracuje z dowolną drukarką dołączoną do systemu.

Program obsługuje się za pomocą menu głównego, zawierającego następujące funkcje:

- **CREATE FILE** (tworzenie tekstu),
- **EDIT FILE** (przeglądanie utworzonego tekstu),
- **VERIFY SPELLING** (program korektorski wraz ze słownikiem angielskim),
- **GLOBAL FORMAT** (kształtowanie parametrów tekstu do drukowania),
- **MAIL MERGE** (program obsługi korespondencji),
- **1 INDEX DRIVE 1** (sprawdzanie zawartości dyskietki znajdującej się w stacji dysków nr 1),
- **2 INDEX DRIVE 2** (sprawdzanie zawartości dyskietki znajdującej się w stacji dysków nr 2),
- **LOAD FILE** (wprowadzanie zbiorów z dyskietki do pamięci komputera),
- **SAVE FILE** (zapisywanie zbiorów na dyskietce),
- **DELETE FILE** (usuwanie zbiorów znajdujących się na dyskietce),
- **FORMAT DISK** (formatowanie dyskietki do przechowywania tworzonych tekstów).

Uruchomienie poszczególnych funkcji uzyskuje się przez naciśnięcie litery inicjującej jej nazwę, przy czym nie jest wymagana obecność dyskietki z programem w napędzie, za wyjątkiem VERIFY SPELLING i MAIL MERGE. W dowolnym momencie pracy istnieje możliwość powrotu do głównego menu przez naciśnięcie ESC.

1. Proces redagowania tekstu

Redagowanie tekstu składa się z trzech zasadniczych części: tworzenia, jego korekty oraz określenia stylu i formy wydruku.

Tworzenie tekstu jest możliwe po wybraniu z menu opcji CREATE FILE. Ekran tekstowy podzielony jest na funkcjonalne części pozwalające pisać za pomocą klawiatury, informować użytkownika o błędach oraz wydawać dyspozycje w formie poleceń lub pytań.

Program stale informuje o ilości wolnej pamięci pozostającej dla obsługującego. W Atari 800 XL użytkownik ma do dyspozycji 12 645 bajtów, natomiast 130 XE daje w sumie ponad 47 tys. bajtów do wykorzystania, tzn. trzy banki po 15 872 bajty. Jeśli standardowa strona maszynopisu formatu A-4, zawierająca 30 linii po 60 znaków maksymalnie, zajmuje średnio 1 500 bajtów pamięci, to łatwo można przekonać się o możliwościach obu modeli.

Ekran tekstowy informuje również o trybie wprowadzanych liter: małych **(LOWERCASE)** lub wielkich **(UPPERCASE)**. Dla zmiany liter małych na wielkie, podczas korekty napisanego tekstu, używa się klawiszy **CONTROL + CAPS**, natomiast standardowo (w czasie pisania) **SHIFT** lub **CAPS**.

Program wyświetla też na ekranie strzałki wskazujące ustawienie tabulatora, szczególnie przydatne w czasie wypisywania kolumn liczb, nazw czy sporządzania zestawień tabelarycznych. Istnieje możliwość dowolnego kształtowania ustawienia tabulatorów przez użytkownika. Dla całkowitej zmiany używamy komend: **CONTROL + TAB** w celu usunięcia aktualnej tabulacji oraz **SELECT + TAB** dla ustawienia tabulacji w miejscu, które uznamy za właściwe. Można też kasować pojedyncze strzałki tabulatora **(START + TAB)**.

Ostatni stały komunikat określa miejsce kursora na ekranie, podając numer kolumny **(C=)** oraz numer linii **(L=)**.

Atariwriter Plus umożliwia redagowanie tekstu o szerokości od 5 do 249 kolumn. Użytkownik może zaplanować liczbę kolumn przed rozpoczęciem pisania bądź po stworzeniu części lub całości tekstu. Wymaga to przesunięcia kursora w trybie tekstowym na początek dokumentu czy artykułu i naciśnięcia klawiszy **OPTION + C**. Na pytanie: **"How many columns?"** wpisujemy wybraną wartość i naciskamy **RETURN**.

Wprowadzanie tekstu jest czynnością bardzo prostą, choć wymaga zapamiętania kilku własności programu. Polecenia wydawane przez użytkownika z klawiatury sygnalizowane są pojawieniem się na ekranie symboli specjalnych, któ-

re nie są odtwarzane na drukarce, a świadczą jedynie o przyjęciu przez program komendy obsługującego.

W czasie pisania specjalna rola przypada klawiszowi **RETURN**. Po jego naciśnięciu kursor przemieszczony zostaje automatycznie na początek nowej linii, stąd używa się go do oznaczenia pojedynczej linii, końca paragrafu lub wprowadzania dodatkowych pustych linii pomiędzy fragmentami tekstu.

Jeśli przystępujemy do pisania dokumentu, listu czy artykułu, stajemy przed problemem umieszczenia tytułu dokładnie na środku strony **(CONTROL + C)** lub przesunięcia fragmentu np. daty do prawego marginesu **(dwukrotne naciśnięcie CONTROL + C)**. Atariwriter Plus realizuje to automatycznie po wydaniu powyższych poleceń na początku tekstu.

Fragmenty tekstu stanowiące pewną logiczną całość wyodrębnia się zaznaczając początek akapitu przez podanie komunikatu **CONTROL + P**. Jeżeli zamierzamy drukować materiał na pojedynczych arkuszach papieru (np. maszynowego w formacie A-4), możemy polecić wykonanie na każdej stronie nagłówka **(CONTROL + H)** lub stopki **(CONTROL + F)**, zawierającej numerację stron **(SHIFT** + 8) albo dowolny tekst nie przekraczający długości dwu pojedynczych linii.

W trakcie pisania można tworzyć w pewnym zakresie styl wydruku, choćby przez oznaczenie podkreśleń (klawisz **inverse video** lub **CONTROL + U** we fragmencie napisanym) czy rozstrzelenia tekstu **(SELECT + E** na początku i końcu fragmentu). W czasie pisania wzorów chemicznych, matematycznych czy przypisów niezbędne jest posłużenie się pismem podwyższonym **(CONTROL + G + 4)** lub obniżonym **(CONTROL + G + 5)**. Atariwriter Plus pozwala na zaprogramowanie wolnych miejsc, zawierających do 35 znaków, celem późniejszego uzupełnienia **(OPTION + INSERT)** lub wręcz zatrzymanie pracy drukarki **(CONTROL + E)** w dowolnym punkcie tekstu. Program umożliwia również automatyczne numerowanie rozdziałów i podrozdziałów w obrębie pisanego zbioru, dzięki komendzie **SELECT + H**.

Podczas tworzenia tekstu zawsze popełnia się błędy, bowiem "najsłabszą" częścią zestawu komputerowego jest jego użytkownik. Wszystkie zauważone pomyłki można usunąć za pomocą klawiszy: **DELETE, CONTROL + DELETE, SHIFT + DELETE**. Komputer jest jednak łaskawy i pozwala na przywrócenie omyłkowo wymazanych linii, zdań przez naciśnięcie **SHIFT + INSERT**.

Jeśli uznamy, że nie mamy już ochoty wpatrywać się w ekran monitora, można zapisać utworzony zbiór na sformatowanej dyskietce i wrócić doń po pewnym czasie. Doświadczeni autorzy wiedzą, że taki proces "dojrzewania" tekstu bardzo pozytywnie wpływa na efekt końcowy. W celu zapisania zbioru **(SAVE FILE)** należy wpisać numer napędu dyskietek, krótką nazwę i nacisnąć RETURN. Jeśli nazwa zbioru wprowadzanego na dyskietkę jest identyczna z już istniejącą, program zapyta czy zastąpić zbiór **(REPLACE EXISTING FILE Y/N?)**. Mamy też możliwość zapisania tylko fragmentu utworzonego tekstu. Jego początek oznaczamy przez **OPTION + B**, a koniec **OPTION + S**. Należy również wpisać nazwę zbioru tak jak poprzednio.

2. Operacje na fragmentach tekstu

Ponowne wprowadzenie zbioru do pamięci komputera umożliwia funkcja **LOAD FILE** z głównego menu. Po komunikacie **FILE TO LOAD** należy wpisać numer napędu dyskietek, nazwę zbioru i nacisnąć RETURN. Jeśli nazwa umknęła nam z pamięci, można dokonać przeglądu zawartości dyskietki funkcjami menu głównego: **1 INDEX DRIVE** lub **2 INDEX DRIVE**. Komputer wyświetli nazwy wszystkich zbiorów znajdujących się na dyskietce. Jeżeli przed wprowadzeniem nowego zbioru w pamięci komputera znajdował się inny tekst, program zapyta czy usunąć go z pamięci **(ERASE FILE IN MEMORY Y/N?)**. Istnieje również możliwość połączenia zbioru będącego w pamięci komputera z dowolnym tekstem zapisanym na dyskietce, jeśli tylko pozwoli na to pojemność pamięci operacyjnej maszyny. Należy zatem ustawić kursor w miejscu dołączenia zbioru z dyskietki, nacisnąć **OPTION + L** oraz wpisać jego nazwę w odpowiedzi na pytanie: **"FILE TO MERGE?"**. Wykonanie potwierdzamy klawiszem RETURN.

Wszystko jest więc przygotowane do ostatecznego zredagowania utworzonego tekstu przed wydrukowaniem. Program stwarza szerokie możliwości w zakresie wymazywania i przemieszczania fragmentów, bez konieczności powtórnego przepisywania. Wszystkie operacje tego typu wykonywane są z wykorzystaniem tzw. bufora bezpieczeństwa **(ang. failsafe buffer)**. Jest to obszar pamięci, do którego przesyłane są fragmenty tekstu celem odzyskania w przypadku zmiany decyzji lub popełnienia błędu. Pojemność bufora bezpieczeństwa wykorzystywanego w operacjach redagowania wynosi 8 000 bajtów dla Atari 130 XE, natomiast w modelach 64 KB zależy od zakresu wolnej pamięci. Zawartość bufora kasowana jest automatycznie po wykonaniu następnej operacji lub przez użytkownika (komunikat **START + E**), jeśli uzna on za niecelowe kontynuowanie przenoszenia, wymazywania lub powtarzania fragmentów.

W czasie redagowania użytkownik programu Atariwriter Plus dysponuje następującymi możliwościami:

- wymazywanie części tekstu,
- przemieszczanie fragmentów tekstu,
- powtarzanie fragmentów tekstu,
- sporządzanie alfabetycznej listy wyrazów lub zdań tworzących zbiór,
- obliczanie liczby wyrazów w zbiorze,
- odszukiwanie i zastępowanie słów.

Operacje te polegają na odpowiednim zaznaczeniu parametrów: początku i końca transformowanej części oraz miejsca odtworzenia.

W pięciu pierwszych wypadkach początek przekształcanego fragmentu należy oznaczyć poprzez **OPTION + B**, natomiast w operacji odszukiwania i zastępowania słów miejscem inicjującym może być początek tekstu **(START + S)** lub koniec tekstu **(SELECT + U)**, ale wtedy kierunek przeszukiwania będzie odwrotny. Wymazywanie lub przemieszczanie części tekstu wymaga oznaczenia przez naciśnięcie **OPTION + DELETE** końca fragmentu, który zamierzamy opracować. Odzyskania wymazanej treści z bufora bezpieczeństwa dokonujemy za pomocą komendy **OPTION + X**, w miejscu, które uważamy za stosowne. Istnieje także możliwość wymazywania tekstu od miejsca oznaczenia **(SELECT + DELETE)** do końca zbioru. Trzeba jednak mieć świadomość ograniczonej pojemności bufora, co może spowodować bezpowrotną utratę zapisanych treści.

Gdy mamy zamiar powtórzyć fragment tekstu, jego koniec sygnujemy komunikatem **OPTION + E**, podczas gdy początek i miejsce powtórzenia tak samo jak w poprzednich operacjach. Obliczanie liczby wyrazów w zbiorze wymaga oznaczenia końca interesującego nas fragmentu poprzez naciśnięcie **OPTION + W**. W celu sporządzenia alfabetycznej listy wyrazów lub zdań w zbiorze po każdym wyrazie lub zdaniu należy nacisnąć RETURN dla rozmieszczenia ich w osobnych listach. Następnie zaznaczamy początek (jak w poprzednich operacjach) oraz koniec listy **(OPTION + A)**.

Interesującą funkcją programu Atariwriter Plus jest poszukiwanie i zastępowanie słów. Istnieją dwie możliwości: zastępowanie globalne - w całym zbiorze **(OPTION + G)** lub pojedynczych wyrazów. Użycie tej funkcji wymaga wpisania poszukiwanego słowa **(SEARCH STRING)**, naciśnięcia **RETURN** oraz **START + R**. Następnie należy wpisać słowo zastępujące **(REPLACE STRING)**, ponownie użyć klawisza **RETURN** oraz komendy **SELECT + S**. Kursor przemieści się w miejsce poszukiwanego wyrazu. Po pojawieniu się komunikatu: **"STRING FOUND"** naciskamy **SELECT + R**, co spowoduje zastąpienie odnalezionego słowa.

Jeśli uznamy, że tworzony tekst nie wymaga już większych korekt redakcyjnych, możemy dokonać przeglądu powstałego zbioru. Oprócz standardowej funkcji przesuwania kursora za pomocą klawisza **CONTROL** wraz ze strzałkami, dysponujemy specjalnymi możliwościami:

- przesuwanie kursora na początek tekstu **(SELECT + T)**,
- przesuwanie kursora na koniec tekstu **(SELECT + B)**,
- przesuwanie kursora na początek następnego lub poprzedniego wyrazu w linii **(SELECT + <--/-->)**,
- przesuwanie kursora na początek linii **(CONTROL + A)**,
- przesuwanie kursora na koniec linii **(CONTROL + Z)**,
- przesuwanie kursora o odległość wskazaną strzałkami tabulacji **(TAB)**,
- przesuwanie kursora o jedną stronę wyświetloną na ekranie **(OPTION + strzałka w górę / / w dół)**.

Jednym z poważniejszych mankamentów programu Atariwriter Plus jest brak stałego widoku strony o parametrach identycznych z gotowym wydrukiem. Istnieje jednak możliwość posłużenia się trybem **PRINT PREVIEW**, pozwalającym wyświetlić na ekranie tekst w formie zbliżonej do wydruku. Wymaga to użycia klawiszy **OPTION + P** w trybie edycji tekstu, przy czym kursor powinien znajdować się na początku zbioru.

3. Kształtowanie parametrów tekstu do wydrukowania

Jest to ostatni etap pracy z tekstem przed ujrzeniem efektów swych wysiłków na papierze drukarki współpracującej z komputerem.

Komendy formatowania kształtu wydruku są dostępne po wybraniu opcji **GLOBAL FORMAT** z głównego menu. Zespół rozkazów umożliwiających dobranie formy drukowanych treści oznaczony jest symbolami literowymi, natomiast parametry wydruku widnieją w postaci cyfr, które mogą być zmieniane przez użytkownika. Dla wpisania nowej wartości komendy należy nacisnąć odpowiadającą jej literę, wpisać żądaną cyfrę i nacisnąć RETURN. Użycie klawisza TAB powoduje powrót do wartości wyjściowych, które program proponuje po wywołaniu opcji **GLOBAL FORMAT**. Oto zestaw komend i wybranych parametrów wydruku:

a) margines górny (T) i dolny (B):
- wartość 1, to 1 cal (2,5 cm),
- wartość 0 daje wydruk ciągły, np. na papierze z rolki,

b) odstęp między paragrafami (D):
- wartość 2, odstęp jednoliniowy,
- wartość 4, odstęp podwójny,

c) rodzaj pisma (G):
- wartość 1, pismo standardowe (10 znaków na cal),
- wartość 2, pismo ściesnione,
- wartość 3, pismo proporcjonalne,
- wartość 4, pismo podwyższone,
- wartość 5, pismo obniżone,
- wartość 6, pismo pochyłe,

d) wcięcie akapitu (I) mierzone jest od lewego marginesu liczbą odstępów wielkości litery

e) wyrównanie prawego marginesu (J):
- wartość 0, brak równania do prawej,
- wartość 1, prawy margines wyrównany,

f) numeracja stron (Q):
- wartość 1 uruchamia numerację,

g) odstęp między liniami (S):
- wartość 2, jedna linia,
- wartość 4, podwójny odstęp,
- wartość 6, potrójny odstęp,

h) długość strony (Y) - wartość ta określa miejsce, w którym ma się zaczynać następna drukowana strona. Liczba wpisywana

> 22

21 <

przez użytkownika nie oznacza zatem ilości drukowanych linii. Dla strony formatu A-4 należy wprowadzić 144.

i) zatrzymanie wydruku (W):
- program zatrzymuje wydruk w chwili osiągnięcia dolnego marginesu (wartość 1),

j) lewy (L) i prawy (R) margines mierzone są w odstępach wielkości liter, począwszy od lewej krawędzi kartki, np. dla formatu A-4: lewy margines = 11 a prawy = 73, wtedy uzyskujemy 62 znaki w linii umieszczone w okolicach środka strony, czyli standard wymagany przez większość redakcji przyjmujących maszynopis do drukowania.

k) drugi lewy (M) i prawy (N) margines - określają parametry dla wydruku dwukolumnowego.

Ostatnim etapem pracy z programem Atariwriter Plus jest wydrukowanie tekstu. Umożliwia to funkcja **PRINT FILE** głównego menu, dając do dyspozycji bogaty zestaw urządzeń drukujących. Jeśli w spisie nie znajduje się drukarka podłączona do komputera, istnieje możliwość wprowadzenia kodów sterujących wewnątrz pisanego tekstu **(CONTROL + 0)** lub stworzenia procedury obsługi nietypowej drukarki. Aby uzyskać tę możliwość, należy załadować program Atariwriter Plus poprzez włączenie komputera oraz trzymanie klawisza **SELECT** do momentu pojawienia się komunikatu: **"CUSTOM PRINTER DRIVE EDITOR"**. Naciśnięcie **SPACE BAR** spowoduje ukazanie się edytora z poszczególnymi funkcjami.

4. Dodatkowe funkcje programu

Atariwriter Plus ma dwie cechy wyróżniające go spośród edytorów tekstu dla Atari.

Pierwszą jest możliwość dokonania automatycznej korekty tekstu z wykorzystaniem gotowego lub własnego słownika wyrazów. Druga - pozwala na stworzenie dodatkowych zbiorów informacji np. adresów, przydatnych w obsłudze korespondencji.

Menu programu korektorskiego zawiera następujące nowe funkcje:

- **CORRECT ERRORS**, odszukiwanie błędów z możliwością poprawiania za pomocą submenu korektorskiego,
- **HIGHLIGHT ERRORS**, wyświetlanie znalezionych błędów w trybie inverse video, bez możliwości korekty,
- **PRINT ERRORS**, sprawdzanie tekstu i drukowanie wyrazów z błędami,
- **DICTIONARY SEARCH**, przeszukiwanie słownika w celu sprawdzenia prawidłowości pisowni wskazanych wyrazów,
- **ADD PERSONAL DICTIONARY**, wprowadzanie słownika utworzonego przez użytkownika z dyskietki do pamięci komputera,
- **FILE PERSONAL DICTIONARY**, zapisywanie na dyskietkę słownika utworzonego przez użytkownika,
- **RETURN TO ATARIWRITER**, powrót do programu zasadniczego edycji tekstu.

Przy korzystaniu z modułu korekty błędów należy pamiętać, że program sprawdza wyłącznie pisownię, nie ingerując w zagadnienia kontekstu użycia czy poprawności stylistycznej, nie poprawia też pisowni dużych i małych liter. Jeśli sprawdzane słowo nie znajduje się w słowniku, program traktuje je jako błąd.

Korzystanie z systemu obsługi korespondencji **(MAIL MERGE)** jest bardzo podobne do posługiwania się bazą danych typu korektorskiego, np. **SYNFILE** + (opis: "Komputer" 10/87). Różnica polega na znacznie mniejszych możliwościach, co zresztą wiąże się z jej przeznaczeniem - jest to typowa elektroniczna książka adresowa. Użytkownik dysponuje następującymi nowymi funkcjami **menu MAIL MERGE:**

- **CREATE FILE** (tworzenie zbioru),
- **EDIT FILE** (wprowadzanie oraz aktualizacja rekordów - po naciśnięciu klawisza START),
- **BUILD SUBST** (indeksowanie zbioru),
- **REPEND FILE** (dołączanie zbioru),
- **PRINT FILE** (drukowanie zbioru).

W celu skorzystania ze zbiorów bazy danych MAIL MERGE należy w tekście tworzonym za pomocą Atariwriter Plus wprowadzić komendę **OPTION + M.** Po pojawieniu się znaku specjalnego wpisujemy numer pola wprowadzanego do zbioru i naciskamy RETURN.

Zakończenie

Wykorzystanie wszystkich możliwości programu Atariwriter Plus wymaga dosyć uciążliwego pamiętania wielu poleceń składających się z kombinacji łącznie używanych klawiszy. Do pełnego opanowania obsługi potrzeba więc wiele wprawy i praktyki. W zasadzie nie jest możliwe sprawne posługiwanie się programem bez dokumentacji, stąd tak obszerny artykuł omawiający ten edytor tekstu.

Stosunkowo niedawno pojawił się na "naszym rynku" inny ciekawy program redagowania tekstów - **STARTEXTER**. Jego twórcy zadbali o to, aby posługiwanie się nim było funkcjonalne, a jednocześnie by spełniał on wymagania bardziej wybrednych użytkowników.



\w domu\

Robert Postowicz

Polskie znaki dla ATARI ST

Atari ST jest komputerem o bardzo dobrych, wszechstronnych możliwościach, a jego niska cena uczyniła go popularnym w Polsce. Dla wielu użytkowników pragnących wykorzystać swoje ST do poważniejszych zadań kapitalne znaczenie ma nie rozwiązany dotychczas w pełni problem posługiwania się polskimi znakami. W niniejszym artykule pragnę przedstawić rozwiązanie opracowane na podstawie standardu przyjętego przez Klub Użytkowników ST, działający przy redakcji "Komputera".

Jakie kody?

Pierwszą rzeczą, którą należy omówić, jest kwestia umiejscowienia polskich znaków w ramach kodu ASCII, czy raczej jego odmiany używanej w ST. Naturalne byłoby tu wykorzystanie jednego ze standardów rozpowszechnionych w świecie IBM-ów, np. tzw. standardu Mazovii, jednak ich wadą jest to, iż... pochodzą ze świata IBM-ów i są mało przydatne dla ST. Np. wspomniany wyżej standard Mazovii (patrz "Komputer" nr 10/87) zabiera znak funta ("L") z zestawu ASCII, a znak ten znajduje się na klawiaturze ST pochodzących z Wielkiej Brytanii. Dodatkowym mankamentem tego wariantu jest fakt, że znaki nie są ustawione w kolejności alfabetycznej.

A więc własny standard, tylko jaki? Patrząc na systemowy zestaw znaków ST łatwo jest zauważyć rzadko w Polsce stosowane znaki hebrajskie, zaczynające się od kodu 194. W tym też obszarze należałoby umieścić polskie znaki.

Wykonując jeszcze ukłon w stronę programistów ostatecznie przyjęto następujący wariant: duże polskie litery (ĄĆĘŁŃÓŚŹŻ) zaczynają się od kodu 193 i następują po sobie w kolejności alfabetycznej. Małe znaki polskie umieszczone są w analogiczny sposób, począwszy od kodu 209. Dlaczego właśnie te kody? Ano policzmy: 65 (duże "A" w ASCII) + 128 = 193 oraz 193 + 16 = 209. A każdy, kto trochę programuje, z liczbami takimi jak 16 i 128 zetknął się na pewno wielokrotnie. Być może praktyczne znaczenie takiego układu nie jest w istocie zbyt wielkie, ale niczego on w standardowym zestawie znaków nie "psuje", a to jest przecież najważniejsze.

Polskie fonty

W stosunkowo najlepszej sytuacji są ci, którzy mają wczesną wersję ST z systemem operacyjnym na dysku (oczywiście abstrahując od ogólnych niedogodności, jakie układ ten stwarza). Możliwa jest bowiem trwała zmiana systemowych fontów, poprzez modyfikację dyskietki systemowej (oczywiście, ze względów bezpieczeństwa wszelkie modyfikacje należy przeprowadzać na kopii dyskietki systemowej). Zanim jednak omówimy zmiany, jakie należy poczynić, parę słów na temat sposobu traktowania fontów przez TOS/GEM - system operacyjny Atari ST.

Teoretycznie biorąc, GEM jest bardzo liberalny, jeżeli chodzi o zestawy znaków i opcja ładowania fontów jest przewidziana systemowo. Odpowiedzialną za ładowanie fontów częścią GEM-u jest GDOS (ang. *Graphic Device Operating System*). Podczas inicjacji GEM-u (w ST dzieje się to zaraz po włączeniu komputera) GDOS szuka pliku o nazwie **ASSIGN.SYS**, w którym oczekuje informacji dotyczących nazw plików, zawierających *drivery* poszczególnych urządzeń zewnętrznych oraz fonty przeznaczone do załadowania dla każdego z tych urządzeń (mam tu na myśli *drivery* implementujące systemowe komendy graficzne dla poszczególnych urządzeń, np. standardowe **FX80.SYS** lub **META.SYS**, przy czym *driver* monitora dla wszystkich rozdzielczości - **screen.sys** - zawarty jest w systemie operacyjnym i nie musi być ładowany z dysku). Tak zaanonsowane fonty są później dostępne dla wszystkich programów i można je wczytywać i wyświetlać za pomocą procedur systemowych (w naszym przypadku mógłby to załatwiać program typu *desk accessory*).

Ten sposób ładowania fontów, teoretycznie jedyny poprawny, jest jednak kłopotliwy, i to z dwóch powodów. Po pierwsze, nie istnieje w tej chwili dobry edytor fontów zapisujący je w formacie oczekiwanym przez GDOS (jedyny jaki widziałem był w języku francuskim i jest praktycznie nie spotykany wśród użytkowników ST w Polsce). Po drugie - i tutaj tkwi prawdziwy problem - z powodów, których trudno w tej chwili dociec (podobno brakowało miejsca w ROM-ie albo Digital Research nie zdążyło z odpowiednią przeróbką na czas) GDOS został pominięty w wersji

TOS-u umieszczonej w pamięci ROM i jest dostępny jedynie jako program typu *terminate-and-stay-resident*. Stąd jest on często dołączany, w folderze "AUTO" (gdyż wykonany musi być jeszcze przed inicjacją GEM-u), do programów, które przewidują możliwość zmiany kroju pisma, takich jak chociażby Degas Elite, Easy Draw czy MS-Write. Wszystko to razem jednak sprawia, że ładowanie fontów w powyższy sposób jest niezbyt wygodne dla użytkownika.

I jeszcze, zanim przejdziemy do konkretów, parę słów na temat formatu fontów akceptowanego przez GEM. Każdy zestaw znaków, oprócz samych danych definiujących ich wygląd, ma "nagłówek", zawierający informacje o rozmiarze znaków, ich ilości, sposobie przeprowadzania operacji graficznych (pogrubianie itp.) na znakach, nazwę fontu i całą masę innych informacji. Istotne jest to, iż w nagłówku znajduje się także adres, z którego będą pobierane dane definiujące znaki (na dysku jest on podany względem początku nagłówka, gdyż fonty podczas ładowania podlegają relokacji). Same zaś dane przechowywane są w sposób następujący: załóżmy, że znaki w definiowanym przez nas foncie mają rozmiar 8x8 punktów, a pierwszym znakiem w zestawie jest litera "A". Pierwszy bajt będzie więc definiował pierwszą od góry "linię skaningową" litery A, ale drugi bajt będzie zawierał definicję górnej linii litery B, a nie kolejnej linii litery A! I tak dalej, dopóki nie dojdziemy do ostatniego znaku, po którym następuje definicja kolejnej "linii skaningowej" - aż do linii ósmej, kończącej definicję. Należy tu jeszcze dodać - co jest ważne w wypadku, gdy znaki mają nietypową szerokość, np. 11 punktów, lub font jest proporcjonalny - że definicje linii skaningowych pierwszego znaku muszą się zaczynać na granicy słowa (adres podzielny przez 2), natomiast definicje linii kolejnych znaków są upakowane bezpośrednio za ostatnim bitem definicji poprzedniego znaku - w ten sposób nie marnuje się prawie wcale pamięci. Zainteresowanych dalszymi szczegółami odsyłam do książki "ST Internals", strony 218-222.

Jak więc należy przeprowadzić modyfikację dysku systemowego? Potrzebne nam będą trzy rzeczy: dobry edytor dysku (np. Disk Doctor lub Michtron Utilities), edytor fontów Olifont od programu ST AD (ewentualnie edytor fontów od Degasa - Olifont nie działa w kolorze) i odrobina cierpliwości. Zacząć należy od zaprojektowania polskich znaków. Ci, którzy mają monitor monochromatyczny, powinni zaprojektować polskie znaki w matrycy 8x16 punktów, zaś użytkownicy monitorów kolorowych w matrycy 8x8 (dla dobrych efektów proponuję wzorować się na znakach systemowych). Można to zrobić na kartce kratkowanego papieru, aby później przeliczyć dane, lepiej jednak posłużyć się edytorem. Stosując to drugie rozwiązanie i edytor Degasa, należy osobno rozrysować pierwszą (identyczną jak zestaw systemowy) i drugą (z polskimi znakami) połówkę fontu (projektując znaki 8x8 należy po prostu opuścić co drugą linię). Po zapisaniu fontu na dysk należy przekształcić dane do postaci oczekiwanej przez GEM. Można przy tym posłużyć się załączonym programem, napisanym w ST-Basicu. Program ten zapisuje na dysku font w odpowiednim formacie. Plik wynikowy dla fontu 8x16 będzie miał 4096, a fontu 8x8 2048 bajtów długości.

Kolejnym krokiem będzie odnalezienie fontów na dysku. W tym celu wgrywamy edytor dysku i na kopii dysku systemowego (w pliku TOS.IMG) odnajdujemy tekst "8x16 system font" (lub "8x8 system font"). Następnie odliczamy 590 z "hakiem" bajtów od początku tego napisu - powinniśmy trafić na sekwencję "07 F8 08 00 00" (można ją także odszukać za pomocą edytora). Ten ostatni, zerowy bajt jest właśnie pierwszym bajtem danych opisujących font. Pozostała już teraz najprostsza, ale i najżmudniej-

> 24

```
; Program podstawiający polskie znaki pod klawisz "Alternate"
; Napisał R. Postowicz, 18-03-88

; Gotowy program z rozszerzeniem ".PRG" wstawić do foldera "AUTO"

; inicjalizacja - zwolnij niepotrzebną pamięć

            move.l  4(a7), a5       ; adres basepage
            move.l  $c(a5), d0      ; długość kodu
            add.l   $14(a5), d0     ; długość danych
            add.l   $1c(a5), d0     ; długość bss
            add.l   #$100, d0       ; + 256 bajtów na basepage
            move.l  d0, l_bajt      ; zapamiętaj całkowitą długość
            move.l  d0, -(a7)
            move.l  a5, -(a7)
            move.w  #0, -(a7)
            move.w  #$4a, -(a7)     ; setblock, GEMDOS
            trap    #1
            add.l   #12, a7         ; korekta stosu

; zainstaluj nową procedurę obsługi przerwania

            move.w  #1, -(a7)
            move.w  #14, -(a7)      ; iorec, XBIOS
            trap    #14
            addq.l  #4, a7          ; adres parametrów bufora w d0
            move.l  d0, bufor_inf   ; zapamiętaj
            move.l  d0, a0          ; teraz w a0
            move.w  6(a0), ost_zapis ; zapamiętaj ostatni indeks zapisu
            move.l  #podstaw, -(a7)
            move.w  #38, -(a7)      ; supexec, XBIOS
            trap    #14
            addq.l  #6, a7

; gotowe, zakończ program lecz pozostaw go w pamięci

            move.w  #0, -(a7)
            move.l  l_bajt, -(a7)
            move.w  #$31, -(a7)     ; keep_process, GEMDOS
            trap    #1

; procedura instalująca przerwanie, wykonana w trybie 'supervisor' MC68000

podstaw:    or.b    #8, $484            ; ustaw bit 4 conterm
            move.l  $118, st_przerw     ; zapamiętaj stary wektor
            move.l  #przerwanie, $118   ; zainstaluj nową procedurę
            rts

; dalej następuje nowa procedura obsługi przerwania klawiatury

l_kod       =       18                  ; liczba zmienianych kodów

przerwanie: movem.l a0-a2/d0-d3, -(a7)  ; zapamiętaj zaw. rejestrów
            pea     adr_powrotny
            move.w  sr, -(a7)           ; symuluj wejście w przerwanie
            move.l  st_przerw, a0
            jmp     (a0)          ; wykonaj starą procedurę (konieczne!)
adr_powrotny move.w sr, -(a7)           ; zapamiętaj rej. wskaźników
            or.w    #$0700, sr          ; zablokuj przerwania
            move.l  bufor_inf, a0       ; adres parametrów bufora
            move.w  6(a0), d0           ; indeks odczytu
            move.w  8(a0), d1           ; indeks zapisu
            cmp.w   d1, d0
            beq     koniec              ; indeksy równe - bufor pusty
            cmp.w   ost_zapis, d1
            beq     koniec              ; nic nie dodane przez starą proc.
            move.w  d1, ost_zapis
            move.l  (a0), a1            ; adres bufora
            move.l  (a1,d1.w), d3       ; kod klawisza w d3
            swap    d3                  ; kbshift i kod skan. teraz w d3.w
            move.w  d3, d2
            and.w   #$0300, d2          ; ktorys Shift wciśnięty?
            beq     nie_ma_shiftu
            or.w    #$0300, d3          ; tak, ustaw bity obu Shift'ów
nie_ma_shiftu lea   stare_kody, a2      ; adres tablicy "gorących" kodów
            moveq.l #0, d2
szukaj      cmp.w   (a2)+, d3
            beq     podst_znak          ; ten kod wymieniamy
            addq.w  #1, d2              ; zwiększ indeks
            cmp.w   #l_kod, d2          ; koniec tablicy?
            beq     koniec              ; na to wygląda
            bra     szukaj
podsc_znak  lea     nowe_kody, a2
            clr.l   0(a1,d1.w)          ; wyczyść 4 bajty
            move.w  d3, 0(a1,d1.w)      ; ** kbshift/skan - mozna pominąć **
            move.b  0(a2,d2.w), 3(a1,d1.w) ; wstaw kod ASCII do LSB

koniec      move.w  (a7)+, sr
            movem.l (a7)+, a0-a2/d0-d3
            rte                         ; powrót z przerwania!

l_bajt      ds.l    1
bufor_inf   ds.l    1
st_przerw   ds.l    1
ost_zapis   ds.w    1

stare_kody  dc.w    $081e,$082e,$0812,$0826,$0831,$0818
            dc.w    $081f,$082d,$0b1e,$0b2e,$0b12,$0b26
            dc.w    $0b31,$0b18,$0b1f,$0b2d
            dc.w    $082c,$0b2c         ; dla ang. i USA klawiatury
            dc.w    $0815,$0b15         ; dla klawiatury niemieckiej

nowe_kody   dc.b    209,210,211,212,213,214
            dc.b    215,216,193,194,195,196
            dc.b    197,198,199,200,217,201

            end
```

23 <

sza część całej procedury - porównanie poszczególnych bajtów definicji fontów i wymiana tych, które się różnią. Przydatne tutaj będzie uprzednie wydrukowanie naszego pliku w postaci szesnastkowej. Ci, którzy nie posługiwali się edytorem fontów, powinni uważnie przeczytać umieszczony powyżej opis formatu fontów i wyliczyć położenie bajtów zawierających definicje znaków o "polskich" kodach oraz dane, jakie należy tam wpisać, "od ręki" (przyda się tutaj kalkulator). Jak widać, proponowana modyfikacja systemu wymaga trochę pracy, ale zrobić wystarczy ją tylko raz i polskie znaki pozostaną w systemie na "całe życie".

A co mają zrobić "pechowcy" z systemem operacyjnym w ROM-ie? Zamiast wymiany kości ROM-u lub używania systemu z dysku (chyba jednak zbyt radykalne rozwiązania) proponuję im napisanie własnego programu ładującego fonty. **(UWAGA: wszystkie prezentowane tutaj i w dalszej części artykułu rozwiązania są sprawdzone i gotowe, DZIAŁAJĄCE PROGRAMY są dostępne za pośrednictwem naszego Klubu ST.** Może to być program typu *desk accessory*, ale nie może to być program uruchamiany w folderze AUTO - dlaczego, o tym za chwilę (choć ktoś, kto wnikliwie przeczytał artykuł od początku, mógłby się już teraz domyśleć).

Na czym ma więc polegać zasada działania naszego programu? Otóż szczęśliwym trafem GEM podczas inicjacji przepisuje nagłówki dwu z trzech fontów systemowych (8x8 i 8x16) do względnie niskiej pamięci RAM. A, jak pamiętamy, w nagłówkach tych znajdują się adresy wskazujące początek definicji odpowiednich fontów! Należy więc wczytać spreparowany przez nas (chociażby w opisany wyżej sposób) zestaw znaków do pamięci, a następnie wymienić adres w nagłówku i zabezpieczyć dane przed zniszczeniem - program powinien pozostać rezydentny po zakończeniu. Idąc dalej, można ten program napisać jako *desk accessory* i zawrzeć w nim procedurę zmieniającą format fontu - możliwe będzie wówczas bezpośrednie wczytywanie różnych fontów, np. z Olifonta lub Degasa, gdy tylko na ekranie widoczne jest menu GEM-u.

Nagłówek w pamięci można odszukać znów dzięki nazwie fontu - szukamy tekstu *"8x16 system font"* w mono i *"8x8 system font"* w kolorze. Dodając do znalezionego adresu "magiczną" liczbę 72 otrzymamy adres wskaźnika danych fontu. I jeszcze kilka uwag, które powinny być pomocne przy pisaniu programu:

- Położenie nagłówków fontów ma prawo się zmieniać (i się zmienia) w różnych wersjach TOS-u. Dlatego jeśli chcemy, aby program działał nie tylko na naszym komputerze, ale i innych, najbezpieczniej jest, aby sam odnajdywał nagłówki, a nie posługiwał się z góry określonymi adresami, znalezionymi uprzednio np. za pomocą monitora pamięci.
- Poszukiwanie nagłówka rozpoczynamy nie niżej niż od adresu $800 (2048). Dane poniżej tego adresu (wektory i zmienne systemowe) są dostępne jedynie w trybie *supervisor* mikroprocesora, a nagłówka na pewno tam nie znajdziemy.
- W pamięci pomiędzy nagłówkiem fontu 8x8 a 8x16 GEM zostawia sporo wolnego miejsca - może tam zostać załadowane nasze "akcesorium", i w rezultacie przy porównywaniu pamięci program może znaleźć "sam siebie" zamiast autentycznego nagłówka 8x16. Przemawia tu przeze mnie gorzkie doświadczenie - przetestowana wersja programu "tajemniczo" odmawiała pracy po skopiowaniu na inną dyskietkę, na której przez przypadek nie było innych akcesoriów, które zapełniłyby lukę pomiędzy nagłówkami... Błąd ten był na tyle trudny do wyjaśnienia, że momentami wydawało mi się, iż albo z komputerem, albo ... ze mną jest coś nie w porządku. Tymczasem problemu można uniknąć chociażby porównując dodatkowo bajty znajdujące się za znalezionym tekstem. Na nazwę fontu w nagłówku przewidziane są 32 bajty, a ponieważ nie są one wszystkie wykorzystane, GEM wypełnia "nadmiar" zerami, co jest raczej mało prawdopodobne w przypadku danych programu (choć mimo wszystko dobrze jest to sprawdzić już po kompilacji!).
- I jeszcze wyjaśnienie, dlaczego program nie będzie działał z foldera AUTO - po prostu byłby on wówczas wykonany przed inicjacją GEM-u, a więc i przed przepisaniem nagłówków do RAM-u. A wpisanie nowego adresu do nagłówka ewentualnie znalezionego w ROM-ie miałoby zaledwie umiarkowane widoki na sukces (wyrażenie oryginalnie użyte przez prasę brytyjską dla określenia szans powodzenia 80-osobowej załogi wojskowej Falklandów przeciwko 140--tysięcznej armii argentyńskiej).

Trzeba jeszcze zaznaczyć, że dokonana w opisany sposób wymiana fontów będzie skuteczna tylko w przypadku programów działających w GEM-ie (generalnie biorąc, programy z rozszerzeniem .PRG lub .APP), ponieważ TOS czerpie fonty prosto z ROM-u. Wchodząc nieco głębiej w system, należy stwierdzić, iż warunkiem działania naszego podstawienia z danym programem jest używanie przezeń procedury GEM-u (a ściślej **VDI**) znanej w C jako **v_gtext()** do wyświetlania tekstu. Na szczęście takich programów jest większość.

Kodowanie klawiatury

Załóżmy więc, że udało się - mamy polskie znaki na ekranie. Niestety, to dopiero jedna wygrana bitwa - aby wygrać wojnę, należy mieć jeszcze możliwość swobodnego wprowadzania znaków z klawiatury, co oznacza konieczność jej przedefiniowania. Istnieją co najmniej dwie metody wiodące do celu - zastosowanie jednej z nich zależy jednak od pożądanego układu klawiatury, dlatego tym problemem zajmiemy się najpierw.

Podczas dyskusji nad układem polskiej klawiatury dla ST, jaka odbyła się podczas jednego z zebrań klubowych, wyszły na światło dzienne dwie "szkoły jazdy". Pierwsza z nich opowiada się za układem możliwie zbliżonym do polskiej normy maszyny do pisania ("klawiatura literacka") i oczywiście zabiera z klawiatury kilka znaków, niezbędnych np. programistom. Drugi wariant polega na tym, że polskie znaki uzyskujemy po wciśnięciu klawisza korespondującej litery łacińskiej, trzymając jednocześnie klawisz Alternate (literę ź instalujemy pod klawiszem "X") - tak zwana "klawiatura programisty".

Przejdźmy do bardziej technicznych szczegółów. Jak wspomniałem, istnieją co najmniej dwa możliwe rozwiązania naszego problemu. Pierwsze z nich, dające się zastosować jedynie do wariantu "literackiego", polega na dostarczeniu systemowi nowych tablic definiujących układ klawiatury. TOS korzysta z trzech takich tablic - jednej dla klawiatury normalnej, drugiej dla klawiatury z wciśniętym Shiftem i trzeciej - dla klawiatury z aktywnym klawiszem Caps Lock. Istnieje procedura systemowa - **XBIOS, funkcja nr 16, Keytbl()** dla użytkowników C - pozwalająca podstawić nowe adresy tablic pod dowolnie wybrane z tych trzech. Stary układ klawiatury można ponownie przywrócić za pomocą funkcji **Bioskeys() - XBIOS nr 24.** Zainteresowanych bliższymi szczegółami znów odsyłam do książki "ST Internals" (str. 67-84, 176, 187, 240 oraz listing na końcu książki zawierają potrzebne informacje).

My zajmiemy się drugim rozwiązaniem. Otóż każde naciśnięcie lub puszczenie klawisza powoduje wygenerowanie przerwania przez MFP (MFP 68901 - ang. *Multi Function Peripheral*, jedna z kluczowych "kości" ST). Zadaniem procedury obsługującej to przerwanie jest przyjęcie kodu klawisza od sterownika klawiatury i umieszczenie odpowiednio przygotowanego, 4-bajtowego pakietu informującego o tym wydarzeniu w buforze klawiatury (o ile przerwanie zostało wygenerowane przez klawiaturę, bo są też inne możliwości). Załączony program w asemblerze wykorzystuje fakt, że istnieje wektor systemowy wskazujący położenie procedury obsługi tego przerwania i podstawia własną procedurę w miejsce systemowej.

Zamiast dokładnego omówienia programu proponuję Czytelnikom dokładne przestudiowanie kodu źródłowego, aby zobaczyć "jak to się robi". Pomocna w tym powinna być, kilkakrotnie już wspomniana, bardzo dobra książka "ST Internals" oraz poniższe uwagi.

- Adresem wektora systemowego dla przerwania klawiatury jest **$118** i dostęp do niego jest możliwy jedynie w trybie *supervisor*. Przedstawiony program wykorzystuje w tym celu systemową procedurę **Supexec()** (XBIOS nr 38).
- Systemowe przerwanie nie tylko obrabia "surowe" dane sterownika klawiatury, ale i nadzoruje myszkę, joystick oraz port MIDI. Dlatego niezbędne jest jego wywołanie w ramach naszej nowej procedury starego przerwania (chyba że chcemy je w całości przepisywać...). Konieczne jest z tego samego powodu również sprawdzenie, czy coś rzeczywiście zostało przez nie dodane do bufora klawiatury.
- Oto format 4-bajtowego pakietu wpisywanego do bufora przez systemowe przerwanie:

bajt #3 (najbardziej znaczący):
Aktualny stan klawiszy Shift/Alternate/Control/Caps Lock i in. w formacie identycznym jak zwracany przez funkcję Kbshift(), BIOS nr 11.

bajt #2:
Kod otrzymany ze sterownika klawiatury, informacja o tym, który klawisz został wciśnięty (por. rys. na str. 84 "ST Internals").

bajt #1:
Zawsze 0.

bajt #0:
kod ASCII wynikający z aktualnej tablicy definiującej układ klawiatury (patrz wyżej) oraz stanu klawiszy Shift/Control/itd.

W buforze klawiatury (danych o nim dostarcza funkcja **Iorec()**, BIOS nr 14) zamieniamy jedynie kod ASCII. Ponieważ jednak programy w różny sposób korzystają z tych danych, dla prawidłowego współdziałania z niektórymi z nich może być niezbędne usunięcie z naszego programu jednej z instrukcji (jest ona zaznaczona w komentarzach programu), wówczas bardziej znaczące słowo pakietu będzie wyzerowane.

- Tym, którzy chcieliby zaprojektować własny układ klawiatury, niezbędna będzie informacja dotycząca kodów klawiatury, znajdująca się w książce "ST Internals" (a gdzieżby indziej), str. 84 a także 154. W programie wystarczy zmienić stałą **l_kod** (jeśli konieczne) oraz zawartość tablic **stare_kody** i **nowe_kody.**

Załączony program został napisany przy użyciu asemblera AssemPro i dla prawidłowej asemblacji za pomocą innego asemblera (względnie kompilatora z *in-line assembly*) mogą być niezbędne drobne kosmetyczne zabiegi, głównie w sferze dyrektyw asemblera (np. zamiana "dc.w" na ".dc.w"). Niestety, nie mogę w tej chwili zagwarantować, iż program ten będzie poprawnie współpracować ze wszystkimi programami (one też mogą przecież mieć swoje zdanie na temat przerwania lub bufora klawiatury), ale szereg z nich zostało już przetestowanych z wynikiem pozytywnym, m.in. 1st Word i 1st Word Plus.

Czy czegoś tu czasem nie brakuje?

Owszem. Brakuje informacji na temat zainstalowania polskich znaków na drukarce. Jest to jednak temat bardziej ogólny, dotyczący użytkowników wszystkich typów komputerów, a rozwiązań jest niemal tyle, ile typów drukarek i często zależą one od konkretnego programu, z którym drukarka ma współpracować. Z tego powodu wymaga on szerszego potraktowania.

\W domu\

Tomasz Mazur

Programiki dla Atari XE/XL

Z programem **Mirosława Matlęgi** spotkaliśmy się już w pierwszym odcinku tego cyklu. Choć trochę szkoda, że "Migający kursor" nie był w pełni jego autorstwa (główna procedura była publikowana w "Atari User"), to jednak chciałbym zaprezentować Państwu inny program nadesłany przez kolegę Mirka. Program, zajmujący oczywiście regulaminowe 15 wierszy, rozwiązuje równania algebraiczne trzeciego stopnia, tj. A * x**3 + B * x**2 + C * x + D = 0, a ponieważ są to typowe obliczenia, nie ma więc potrzeby opisywania jego budowy. Programy takie pisze się przede wszystkim wtedy, gdy mamy do rozwiązania duże i pracochłonne problemy. Najpierw piszemy algorytm (schemat) obliczeń, a następnie zamieniamy go na język zrozumiały dla komputera (Basic, Asembler, Pascal itp.). Zazwyczaj do obliczeń tego typu używa się niewielu instrukcji (w przypadku Basica), poza wprowadzaniem danych (INPUT) i wyprowadzaniem danych (PRINT), mamy oczywiście instrukcje matematyczne, instrukcje warunku IF ... THEN, skoków do określonego wiersza (GOTO) czy podprogamu (GOSUB) ewentualnie pętle upraszczające obliczenia FOR ... TO ... STEP ... NEXT ...

WIELOMIAN

```
1 ? CHR$(125): ? "Podaj wspolczynniki rownania ":
? "wg. malejących potęg": ? : INPUT A,B,C,D:
W=-B/(3*A) : PI=3.14159
2 P=(3*A*W^2 + 2*B*W + C)/A : Q=(A*W^3+B*W^2+C*W+D)/A : WR=W^2/4+P^3/27
3 IF WR>0 THEN 5
4 ? : ? : GOTO 7
5 ZAJ=SGN(-Q/2+SQR(WR)) : ZAK=(ABS(-Q/2+SQR(WR)))^(1/3) :
ZA=ZAJ*ZAK : ZBJ=SGN(-Q/2-SQR(WR))
6 ZBK=(ABS(-Q/2-SQR(WR)))^(1/3) : ZB=ZBJ*ZBK : Z1=ZA+ZB :
? "X1 = "; INT((W+Z1)*100) /100 : END
7 IF WR<0 THEN 9
8 GOTO 15
9 Y=3*Q/(2*P*SQR(-P/3))
10 IF Y>0 THEN F1=ATN(SQR(Y^(-2)-1))
11 IF Y=0 THEN F1=PI/2
12 F1=PI+ATN(SQR(Y^(-2)-1)) : ? "X1 = ";
INT(W+2*SQR(-P/3)*COS(FI/3)*100) /100
13 ? "X2 = "; INT(W+2*SQR(-P/3)*COS(FI/3+(2*PI)/3)*100)/100
14 ? "X3 = ";INT(W+2*SQR(-P/3)*COS(FI/3+(4*PI)/3)*100)/100:
END
15 ? "X1 = ";INT(W-2*(Q/2)*(1/3)*100)/100 :
? "X2=X3 = ";INT(W+(Q/2)^ (1/3)*100)/100
```

Na zakończenie już prawie tradycyjnie niespodzianka. Tym razem program wygrzebany z jednego ze starych numerów amerykańskiego miesięcznika, poświęconego komputerom firmy Atari, "Antic". Krótka procedura napisana w języku maszynowym, a uruchamiana z Basica, pozwala na uzyskanie słynnej już tęczy. Odpowiednio wkomponowana we własny program da nam znakomite efekty graficzne. Linie od 0 do 50 służą wpisaniu procedury do pamięci począwszy od adresu 1664 oraz określają wartość 'X', o której za chwilę. Linia 60 to oczywiście główna procedura, zapisana pod instrukcją DATA. Należy zwrócić uwagę na wartość 'X', określa ona bowiem "zakres działania tęczy". Sprawdzając różne wartości udało mi się uzyskać kilka ciekawych efektów:

X=5 - mieniące się tęczowo litery,

X=6 - tęcza na całej powierzchni ekranu,

X=8 - migająca ramka ekranu (tak jak w listingu),

X=9 - efekt wybuchu, eksplozji.

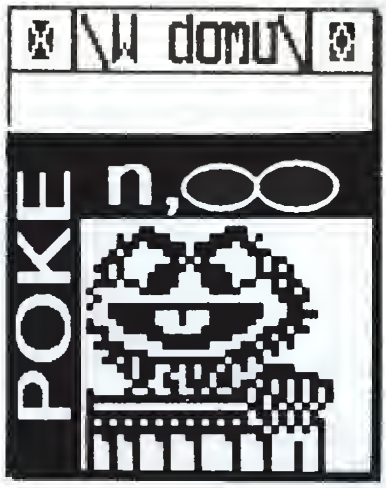
Oczywiście sprawdziłem tylko kilka wartości, dalszą zabawę pozostawiając Czytelnikom.

Przypominam prośbę o dokładne sprawdzanie poprawności nadsyłanych programów i chociaż krótki opis. Niektóre z nich zapowiadają się znakomicie, a potem okazuje się, że popełnione np. w czasie przepisywania błędy nie pozwalają na ich uruchomienie.

Teraz już pora na "tęczowy program".

RAINBOW

```
10 CT=0: X=8
20 CT=CT+1: READ Q: IF Q=-1 THEN 40
30 POKE 1663+CT,Q: GOTO 20
40 Z=USR(1664,X)
50 END
60 DATA 104,104,104,
168,232,142,10,212,138,153,18,
208,169,6,205,31,208,208,241,
196,-1
```



ZX Spectrum

Marcin Orłowski z pierwszej klasy VII LO w Szczecinie prosi o kontakt z innymi "włamywaczami" (adres: Radomska 38, 71-002 Szczecin).

List Marcina otwierają modyfikacje do **SABOTEUR** II firmy Durell. Zatrzymanie upływu czasu w grze daje nam POKE 37122,0. Pod tym adresem powinna być liczba 53 (zmieniamy DEC(HL) na NOP). Nieśmiertelną "nieśmiertelność" daje nam POKE 61340,201, gdzie powinno być 62 (zmiana LDA,#n na RET). Poprawki wpisujemy do głównego bloku (start 25100, długość 40436). W COPY COPY wgrywamy ten segment rozkazem LOAD AT 24083 (odejmujemy 17 na nagłówek). Gra ma kilka poziomów, które możemy osiągnąć po wpisaniu odpowiedniego hasła:

2 - JONIN
3 - KIME
4 - KUJI KIRI
5 - SAIMENJITSU
6 - GENIN
7 - MI LU KATA
8 - DIM MAK
9 - SATORI

Dla niecierpliwych Marcin podaje rozwiązanie. W grze należy na dziewiątym poziomie odnaleźć 14 kodów (są wyświetlane pod liczbą punktów), zanieść je do "szyfrownika" obok rakiety i wyłącznikiem po drugiej stronie rakiety odciąć dopływ prądu do ogrodzenia (FENCE IS OFF). Zaczynamy zabawę od dziewiątego poziomu (SATORI) i czekamy aż nasz bohater sam zeskoczy z lotni, a następnie idziemy: raz w dół, raz w lewo, raz w dół, dwa razy w prawo, raz w dół, trzy razy w lewo, skaczemy w dół i odnajdujemy "żywe" drzewo. Dalej trzy razy w prawo, dwa razy w dół, trzy razy w lewo, dwa razy w dół, trzy razy w lewo, cztery razy w dół, raz w prawo i naciskamy "Fire" potrzebną ilość razy. I to tyle, podobno działa. Na zakończenie opisu tajemnicze ostrzeżenie, by nie "dochodzić do końca ekranu z prawej strony". Mam nadzieję, że ci, którzy widzieli tę grę, będą wiedzieli o co chodzi - mnie ta przyjemność ominęła.

Następną poprawianą przez Marcina grą jest **EXOLON** firmy Hewson. Wpisujemy:

```
100 LOAD " "CODE: POKE 64532,190: RANDOMIZE USR 64512: FOR f=23296 TO 23325: READ a: POKE f,a: NEXT f: POKE 65105,195: POKE 65106,0: POKE 65107,91: RANDOMIZE USR 65082
110 DATA 33,47,248,17,255,255,1,160,146,237,184,49,94,109,251,62,0,50,29,157,175,50,110,131,50,80,146,195,96,109
```

Powyższy program załaduje grę, wprowadzi poprawki i wystartuje (wersja, dla której będzie działał, jest następująca: loader w Basicu, segment z nagłówkiem 64512,768 oraz dwa segmenty bez nagłówka o długościach 4096 i 37536).

Ostatnia poprawiana przez Marcina gra to **SLAP FIGHT** (składająca się m.in. z segmentów SCREEN,50000,5452 i FIGHT,25500, 39020). W COPY COPY wpisujemy poprawki dające nam "nieśmiertelność":

POKE 48872,0, POKE 48873,0, POKE 48874,0 lub "nietykalność":

POKE 57175,201.

Ten sam efekt uzyskamy po wpisaniu własnego programu ładującego:

```
10 CLEAR VAL"24999": BORDER NOT PI:
POKE VAL"23743", VAL"80"
20 LOAD" " CODE VAL"5E4":
RANDOMIZE USR VAL"5E4"
30 LET A=NOT PI: LET B=VAL"48872": LOAD" "CODE
40 POKE B,A
50 POKE B+1,A
60 POKE B+2,A
70 POKE VAL"57175",VAL"201":
POKE VAL"23743",VAL"88"
80 RANDOMIZE USR VAL"64305"
```

Marcin Nowicki z Wrocławia powrócił do opisywanej już gry B.C.Bill firmy Imagine Software. Jest to zabawny żart na tematy jaskiniowe z interesującym motywem muzycznym (podaliśmy już sposób na wydobycie tej muzyki). Marcin podaje jak poradzić sobie w życiu bez powiększania rodziny. Wpisujemy po prostu własny program ładujący:

```
10 PAPER 7: INK 7: BORDER 0: CLS: PRINT AT 20,0: LOAD" " SCREEN$: PRINT AT 20,0: LOAD" " CODE
20 POKE 47589,201: RANDOMIZE USR 25856
```

Bartosz Mielczarek z Łodzi, uczeń Szkoły Podstawowej Nr 28, nadesłał poprawki poprawek do **JACK THE NIPPER**, gdyż podane poprzednio nie działały na jego czterosegmentowej wersji gry. Do tej wersji należy wpisać i uruchomić nowy program ładujący:

```
10 CLEAR VAL"24791": LOAD" " CODE: LOAD" " CODE: RANDOMIZE USR VAL"24830": POKE VAL "44278", VAL"58": POKE VAL"44285", VAL"58": LOAD" " CODE VAL"16464": RANDOMIZE USR VAL"24833"
```

W podobny sposób ułatwimy sobie zadanie w **LIGHT FORCE**:

```
10 CLEAR VAL"24791": LOAD" " CODE: LOAD" " CODE:
RANDOMIZE USR VAL"24830":
POKE VAL "40721", VAL"201":
POKE VAL"40725", NOT PI: LOAD" " CODE VAL"16464":
RANDOMIZE USR VAL"24833"
```

Ostatnie nadesłane przez Bartosza poprawki przeznaczone są dla **SAMANTHA FOX STRIP POKER** (ze względu na jego młody wiek nie podam efektu ich działania)

```
10 LOAD " " SCREEN$
20 FOR F=20728 TO 20735
```

> 26

25 <

30 READ A : POKE F,A
40 NEXT F
50 DATA 62,6,50,112,91,195,
224,81

Atari XE/XL

Dariusz Jankowski (członek KMK) odnalazł sposób na wybranie dowolnego z 10 poziomów w **MINER 2049'er.** Trzeba w programie demonstracyjnym odczytać i zapisać numer telefonu firmy, łącznie z kodem (213). Następnie po wystartowaniu gry, na pierwszym poziomie, należy wpisać ten numer w całości. Od tej chwili naciśnięcie SHIFT i cyfry od 2 do 0 przeniesie nas do poziomu 2 - 10.

Tomasz Hryszko z Białegostoku, lat 14, napisał jak powiększyć limit błędów w **SPACE INVADERS.** Metoda, którą proponuje, to wgranie programu monitorującego (Tomek używa ASM), a następnie wczytanie SPACE INVADERS od adresu 21133 (dziesiętnie). Modyfikujemy zawartość adresu 22296 wpisując nowy limit błędów, tzn. liczbę z zakresu 1,...,255. Zmieniony program nagrywamy na taśmę. Poprawka ta ma pewne działanie uboczne, jednak bez wpływu na grę. W nowej wersji zamiast liczb określających naszą sytuację pojawiać się będą znaki graficzne.

Tomasza Wierzbickiego, ucznia trzeciej klasy Technikum Łączności ze Szczecina, poznaliśmy już wcześniej. Do odnajdowania i wprowadzania poprawek używa on programu UMON XL (spis rozkazów tego monitora jest dostępny po naciśnięciu HELP). Można użyć innego programu monitorującego, lecz takiego, który pozwala na wybór adresu ładowania. Tomek ostrzega przed stosowaniem disasemblera emitowanego w Radiokomputerze, gdyż wprowadza on błędy. Podane numery sektorów dotyczą wersji dyskowych, gdzie modyfikacje wprowadzamy do kopii (najlepiej na nowy dysk). W poniższym zestawieniu liczby podawane są heksadecymalnie, każdemu adresowi towarzyszy wartość, którą należy wpisać, by uzyskać nieśmiertelność w grze.

Wszystkie programy wgrywamy od adresu 4000 (hex)! Posiadacze magnetofonów omijają program ładujący (jeżeli istnieje dla danego programu).

POPEYA - wgrywamy od sektora 04-60
665A, EA
665B, EA
665C, EA

DAN STRIKES BACK - od sektora 04-60
6174, EA
6175, EA
6176, EA

DROP ZONE - od sektora 04-90
limit pocisków: 7C50, EA
7C51, EA
7C52, EA
limit błędów: 5049, EA
504A, EA
504B, EA

MONTEZUMA'S REVENGE - od 04-10
467A LDA #$NN - limit błędów
(467B, FF - maksymalny limit 255)

PASTFINDER - od 04 do 50
5C51, EA
5C52, EA
5C53, EA

Dodatkowy wehikuł uzyskujemy po zebraniu szczątków zniszczonych instalacji wroga.

ELECTRICIAN - wgrywamy od 04 do 90. Sprawdzanie limitu błędów wygląda następująco:
7020 LDA #$06 - ustalenie limitu
7022 STA $08C5 - zapamiętanie pod 02C5
i dalej
7E87 LDA $08C5 - odczytanie zawartości
7E8A SBC #$01 - zmniejszenie o 1
7E8C STA $08C5 - zapamiętanie po 08C5

Nieograniczony limit błędów uzyskamy przez
7E8B, 00

RIVER RAID - od 04 do 10, zwiększamy zapasy paliwa:
4482, EA
4483, EA

WARCOPTER - wersja dyskowa
zmieniamy limity rakiet i naboi - sektory 04-60
6355, EA
6356, EA
6357, EA - naboje
665B, EA
665C, EA
665D, EA - rakiety
zmieniamy limit błędów - SEKTORY 70-100
7728, EA
7729, EA
772A, EA

SATAN'S HOLLOW
4047, FF - 255 istnień

W THEATRE EUROPE na pytania komputera odpowiadamy następująco:
ENTER AUTHORISATION CODE piszemy MIDNIGHT SUN (broń jądrowa),
ENTER CHEMICAL CODE - wpisujemy CAPITAL CITY (broń chemiczna).

Grzegorz Czapkiewicz

Commodore 64

BOULDER DASH

Boulder dash firmy First star software stał się na naszym rynku prawdziwym przebojem. Przez wiele miesięcy utrzymywał się na pierwszym miejscu na listach przebojów. Opisywać nie trzeba, gdyż każdy z nas już na pewno w nią grał. Powiem tylko, że naszym zadaniem jest zebranie odpowiedniej liczby diamentów i dotarcie do wyjścia. Poprawimy teraz nasze programy. Podane przeze mnie POKE'i dotyczą **BOULDER DASH'a I i III.** Wpisujemy: POKE 9785,80 - kamienie, które na nas spadną, nie zabijają nas. POKE 9832,0:9842,0:9854,0:9864,0 - uodporniają nas na duchy.

FALCON PATROL

Następną grą z serii zręcznościowych będzie **FALCON PATROL** firmy Virgin games. Wcielasz się w rolę pilota samolotu odrzutowego. Twoje zadanie to obrona lotnisk przed atakami wrogich odrzutowców. Na tych samych lotniskach możesz zaopatrywać się w amunicję i paliwo do samolotu. Na jednej planszy jest sześć lotnisk, strata ich kończy się rozbiciem samolotu. Jeżeli chcesz sobie postrzelać dłużej, musisz wpisać POKE 16764,36: 16705,2:SYS 6640

Marcin Topolewski

W domu



Program: *MARBLE MADNESS*
Producent:
Rok produkcji: *1987*
Komputer: *Atari ST*
Cena: *19,95 GBP*

"Szalona kulka" zawitała w końcu na ekrany ST. Mimo zaawansowanego (jak na gry komputerowe) wieku - premiera odbyła się w 1985 dla 8-bitowego Atari - MARBLE MADNESS prezentuje się nadal bardzo dobrze, a nawet jakby odmłodniała dzięki szybkiej, kolorowej grafice. Koncepcja gry dziś nie jest już przełomowa i tak atrakcyjna, by przebojem zdobyć rynek, ale pozostałe walory powinny pomóc "Kulce" wtoczyć się na wysokie miejsca list przebojów.

Zasada gry jest bardzo prosta. Musimy w ograniczonym czasie przetoczyć naszą kulkę do mety w każdym z kolejnych etapów. Poruszamy się po dziwnych trójwymiarowych płaszczyznach, połączonych kładkami, zwodzonymi mostami i rynnami. Po drodze musimy omijać różne nieruchome obiekty oraz unikać spotkania z innymi "wrogimi" kulkami, wędrującymi sześcianami i walcami. Czasem pozornie stała powierzchnia może zacząć falować lub zza ściany niespodziewanie wychyli się trąba odkurzacza i zdmuchnie naszą kulkę z wąskiej kładki. Na szczęście mamy nieograniczoną liczbę kulek i po chwili na planszy pojawia się następna. Zabiera to jednak cenny czas, a czas właśnie limituje nasze poczynania. Każdy etap jest trudniejszy od poprzedniego i moim jurorom udało się dojść tylko do szóstego etapu.

Zalety gry to kolorowa grafika, choć obraz przedstawiony jest prostą kreską. Płaszczyzny są pokratkowane, co zwiększa wrażenie trójwymiarowości. Autorzy wykazali wiele pomysłowości w kreowaniu świata gry. Teren jest bardzo urozmaicony, często trzeba wspinać się pod górę lub precyzyjnie prowadzić kulkę po bardzo wąskich kładkach. Liczne pochylnie, katapulty i rynny pozwalają na



przenoszenie się na powierzchnie oddzielone przepaścią. Obrazy przesuwane są bardzo płynnie, bez zachwiania wyrazistości rysunku lub przebarwień.

Akcji towarzyszą odgłosy uderzeń i dość marna muzyczka, która po pewnym czasie staje się denerwująca. Mamy wybór sterowania grą za pomocy myszy lub joysticka i choć sterowanie joystickiem wydaje się łatwiejsze, obydwa sposoby są zbyt trudne do opanowania, by MARBLE MADNESS można było nazwać relaksem. Ponadto możemy zmieniać czas reakcji myszy lub joysticka oraz wybrać grę dwuosobową, co może uatrakcyjnić zabawę. W sumie gra prezentuje się bardzo dobrze, ale kłopoty w opanowaniu sterowania kulką obniżają ocenę. Brak jej też tego czegoś, co powoduje, że gra wciąga i mimo niepowodzeń chętnie do niej wracamy.

Program sprzedawany jest na dwóch dyskietkach w estetycznym opakowaniu zapożyczonym od płyt kompaktowych. Posiadacze ST z jedną tylko stacją dysków muszą na żądanie zamieniać dyskietki, zaś szczęśliwcy z dwiema stacjami mogą włożyć obydwie jednocześnie, a program sam znajdzie potrzebny zbiór.

Program: *TEST DRIVE*
Producent: *ACCOLADE-Electronic Arts*
Rok produkcji: *1988*
Komputer: *Atari ST*
Cena: *19,95 GBP*

TEST DRIVE to wspaniała zabawa w prowadzenie szybkich, sportowych samochodów po krętych, górskich drogach - samochodów o jakich możemy tylko marzyć: Lamborghini Countach, Porsche 911 Turbo, Ferrari Testarossa, Lotus Esprit Turbo i Chevrolet Corvette.

Wkładamy dwie dyskietki do dwóch stacji dysków (lub zamieniamy je w czasie ładowania) i przygotowujemy się do jazdy. Do ręki dostajemy kluczyki z breloczkiem ACCOLADE i stajemy przed trudnym wyborem bolidu, który chcemy prowadzić. Nie dokonujemy go "w ciemno", gdyż program przedstawia sylwetkę każdego samochodu z dość wyczerpującą metryczką. Parametry silnika, moc i przyspieszenia, rodzaj i wielkość opon, a nawet wykres charakterystyki silnika. Ma to pewne walory edukacyjne, choć przypuszczam, że niejeden z młodych fanów motoryzacji mógłby podać znacznie więcej danych o każdym z tych samochodów.

Wybór mamy już za sobą i siadamy za kierownicą. Przed sobą widzimy zestaw wskaźników kontrolnych i drogę przed samochodem. Tu po raz pierwszy zauważamy wierność, z jaką autorzy oddali warunki jazdy i nasz samochód. Każdy z nich ma deskę rozdzielczą dokładnie odpowiadającą rzeczywistości. Ponadto wewnętrzne lusterko wsteczne i dźwignię zmiany biegów, również będące repliką oryginału (stwierdzam to na podstawie zdjęć wybranych przez moją pociechę, rozmiłowaną w motoryzacji).

Uruchamiamy samochód i rozpoczynamy szaleńczą jazdę. Gaz do dechy, zmiana biegów przy najwyższych obrotach silnika i z rykiem oraz piskiem opon ruszamy

do przodu (musimy jednak uważać, by nie przekroczyć maksymalnych obrotów silnika). Prowadzimy bardzo szybką maszynę, więc osiągnięcie prędkości powyżej stu mil na godzinę nie sprawia większego kłopotu. Mój dziesięcioletni syn nie ma utrwalonych nawyków przestrzegania przepisów ruchu drogowego i jeździ z prędkościami powyżej 150 mph (*miles per hour* - mile na godzinę) osiągając średnio powyżej 120 mph. Jest to pierwsze z poważnych wykroczeń drogowych w tej zabawie, ale dzieciom to nie przeszkadza.

Na drodze spotykamy innych uczestników ruchu, jadących w obydwu kierunkach. Dodatkowym utrudnieniem (przypominam - jedziemy górską drogą) jest skała z jednej a przepaść z drugiej strony. Każda kolizja kończy się popękaną szybą i stratą czasu. W miarę jazdy przybywa więcej innych samochodów, a droga staje się bardziej kręta. Po przejechaniu pełnego okrążenia zatrzymujemy się na stacji benzynowej, gdzie uzupełniamy paliwo (bez kartek) oraz jesteśmy informowani o średniej prędkości przejazdu i punktacji. Towarzyszą tej informacji komentarze w rodzaju: "twoja babcia pojechałaby lepiej" lub "uważaj, bo spalisz opony", gdy jedziemy zbyt szybko i zakręty pokonujemy z piskiem opon.



Przy drodze ustawione są liczne znaki, z których większość to ograniczenia prędkości. Policja oczywiście pilnuje przestrzegania tych ograniczeń i szybko wkracza do akcji. W tej grze jednak wszystkie samochody wyposażone są w bardzo pożyteczne urządzenie - wykrywacz radaru. Możemy więc zwolnić prędkość jazdy zanim zostaniemy namierzeni lub ... próbować uciec (to drugie rozwiązanie jest preferowane przez dzieci bez odpowiednich nawyków drogowych, często z powodzeniem). Jeżeli jednak zlekceważymy jednocześnie znak ograniczenia prędkości i wskazania wykrywacza radaru i dogoni nas radiowóz policyjny (zbliżanie się radiowozu możemy obserwować w lusterku wstecznym), zostaniemy zatrzymani i zapłacimy mandat. Tu ostrzeżenie: próby ucieczki po zatrzymaniu kończą się niefortunnie - uderzeniem w radiowóz, aresztem i natychmiastowym zakończeniem gry.

Autorzy zadbali o zachowanie realiów. Wnętrza samochodów i warunki jazdy odpowiadają rzeczywistości. Drogi są kręte, musimy mijać lub wyprzedzać inne samochody. Duże wrażenie robi pokonywanie wzniesień. Zabawnym uzupełnieniem są muchy rozbijające się co pewien czas na przedniej szybie. Napotykamy innych uczestników ruchu i widzimy jak się zbliżają, a następnie w lusterku wstecznym obserwujemy malejącą sylwetkę (przy naszym stylu jazdy nikt nas nie wyprzedza).

Tu kończą się realia, a zaczyna joystick i klawiatura. Trzeba treningu, by pozbyć się odruchu szukania pedałów pod stolikiem i prób kręcenia nieistniejącą kierownicą. Okazuje się jednak, że można sobie poradzić tylko za pomocą joysticka. Ruchy w prawo lub w lewo powodują skręty samochodu (na ekranie widzimy obracającą się kierownicę), w przód - przyspieszenia, w tył - zwolnienie. Naciśnięcie "Fire" przy joysticku wychylonym do przodu zmienia bieg na wyższy, przy cofniętym - pozwala na redukcję. Każdej zmianie biegów towarzyszy obrazek dźwigni (po naciśnięciu D dźwignia zmiany biegów będzie stale widoczna). Dodatkowo klawisz O spowoduje, że ruchy joysticka (łącznie z "Fire") będą odpowiadały realnym kierunkom zmiany biegów. Możemy też zatrzymać grę klawiszem P, wyłączyć muzykę za pomocą M, zaś S pozbawi nas efektów dźwiękowych. Ponowne naciśnięcie każdego z tych klawiszy przywróci poprzedni stan. W rozpaczliwej sytuacji możemy wcisnąć Cntrl-R i rozpoczniemy zabawę od początku.

Gra ta wzbudza wielkie emocje u młodych kierowców (dorośli jednak dość szybko się nudzą) i mimo kilku braków edukacyjnych (preferuje poważne wykroczenia drogowe) zasługuje na najwyższe oceny. Doskonała grafika i dźwięk, precyzja wykonania powodują, że TEST DRIVE jest jedną z najlepszych gier dla ST.

Grzegorz Czapkiewicz

Program: *GARY LINEKER'S SUPERSTAR SOCCER*
Producent: *Gremlin Graphics*
Rok produkcji: *1987*
Komputer: *Amstrad 464/6128*

SOCCER to amerykańskie określenie piłki nożnej. Gary Lineker to brytyjska gwiazda footballu gwarantująca swym podpisem, że przy tej grze spędzisz niejedno popołudnie.

SUPERSTAR SOCCER umożliwia:

- wystawianie na sprzedaż zawodników i ich werbowanie,
- monitorowanie wyników Twojej drużyny i innych zespołów,
- zapisywanie na taśmie/dysku osiągniętych wyników,
- no i wreszcie samą grę w piłkę nożną.

Jeśli zaczynasz od transakcji handlowych, w których towarem są zawodnicy, pamiętaj, że masz do dyspozycji 250 punktów i przejrzyj najpierw wyniki (**"View options"**).

Każdy gracz ma swoją wartość mierzoną punktami od 0 do 90. Wartość ta oznacza jego umiejętności i możliwości piłkarskie, a więc szybkość, celność strzałów, poziom gry w defensywie. Od Ciebie, jako menedżera, zależy wszystko. Określonego gracza możesz włączyć do swojej drużyny bądź umieścić jego nazwisko na liście transferowej (opcje **"Recruit a player"** i **"A player trade"**). Możesz też raz w sezonie wysłać drużynę na obóz kondycyjny (opcja **"Improve team"**), określając ile czasu ma tam spędzić na treningu. Ale pamiętaj - każda sekunda treningu to jeden punkt mniej. Od Ciebie zależy, na jakiej pozycji będzie grał każdy zawodnik i kto będzie rezerwowym (opcja **"Set up team"**). Nazwiska graczy na właściwej pozycji zaznaczone są białym kolorem, na niewłaściwej - żółtym. Na czerwono oznaczono nazwiska zawodników kontuzjowanych. Opcja **"New leage"** umożliwia wybór między ligą brytyjską i amerykańską oraz określenie, ile spotkań rozegranych zostanie w jednym sezonie - 7 czy 15.

Przed przystąpieniem do meczu możesz przyspieszyć, nawet 10-krotnie, upływ czasu odmierzanego przez boiskowy zegar oraz określić, kto - Ty czy komputer - kontrolować będzie ruchy środkowego napastnika oraz decyzje trenera. Ty decydujesz ponadto, czy przebieg gry (w przypadku gdy ruch zawodników kontroluje komputer) będzie wyświetlany na monitorze. Wynik i tak możesz obejrzeć po meczu. Jeśli wybrałeś "ręczne sterowanie" zespołem, czyli obok napisu "coach" określiłeś nazwę "joystick" - możesz na bieżąco ustalać taktykę ofensywnej i defensywnej gry swego zespołu. **"Coach screen"** (ekran trenera) pojawiający się podczas każdej przerwy w grze umożliwi Ci zmianę taktyki lub zamianę zawodnika. Zwróć uwagę, że ofensywny wariant gry przewiduje trzy różne style. **"Schoot"** - to oddawanie strzałów na bramkę przeciwnika z każdej pozycji, nawet z połowy boiska. **"Pass"** - to gra na centralnego napastnika i adresowanie do niego licznych podań w nadziei na zdobycie gola. Wreszcie **"Mix"** - obejmuje podania i strzały z najbardziej dogodnych pozycji.

Wybierając wariant defensywny możesz zamurować dostęp do własnej bramki (kiedy chcesz utrzymać wynik - opcja **"Defend"**) lub dokładnie kryć przeciwnika uniemożliwiając mu rozegranie piłki (opcja **"Cover"**). I w tym przypadku masz możliwość wyboru złotego środka, czyli opcji **"Mix"**.

Jeśli chcesz wymienić w trakcie gry zawodnika - wybierz opcję **"Substitutions"**, przeanalizuj maksymalne i bieżące możliwości każdego gracza (MP - ***maximum power***=90, EP - ***effective power*** w miarę upływu czasu maleje).

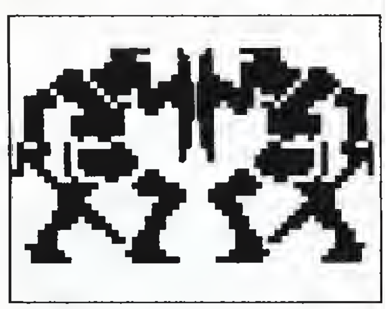
W trakcie gry, jeśli to Ty, a nie komputer, kontrolujesz ruchy środkowego napastnika, możesz podawać piłkę, strzelać i dryblować. Aby podać piłkę, określ ruchem joysticka, do którego zawodnika ją kierujesz i naciśnij "Fire". Jeśli Twoja drużyna atakuje z lewej na prawą, ustawiając joystick w pozycji "góra" i naciskając "Fire" podasz piłkę do najbliższego obrońcy lub skrzydłowego - który będzie bliżej. Pozycja "prawy skos-góra" - to lewy skrzydłowy, prawo - najbliższy skrzydłowy, "prawy skos-dół" - prawy skrzydłowy. Z lewej strony pozycje zajmować będą oczywiście obrońcy. Przy zmianie boisk po przerwie jest odwrotnie.

Strzelając na bramkę przeciwnika naciśnij przycisk "Fire" i zwróć uwagę na znajdujący się pod zegarem wskaźnik. Jeśli znajdować się on będzie blisko strzałki "w górę" i zwolnisz przycisk - strzelać będziesz w róg bramki bliższy górze ekranu. Im bliżej środka wędrować będzie wskaźnik, tym bliżej środka bramki wyląduje Twój strzał.

Jako trener przyjmij, proszę, od nas jedną radę. W przeciwieństwie do Wojciecha Łazarka odwołuj się raczej do młodych zawodników. Tych najbardziej zasłużonych możesz zawsze włączyć do gry w trakcie meczu. Na pewno będzie z nich wtedy większy pożytek.

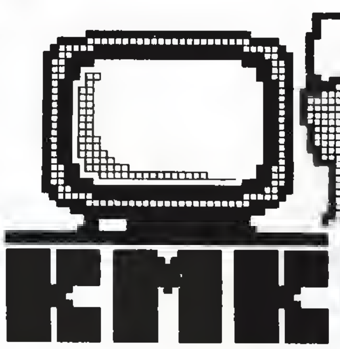
Program: *PHANTOM CLUB*
Producent: *Ocean Software*
Rok produkcji: *1987*
Komputer: *Amstrad 464/6128*

Nawet jeśli lubisz gry, które na własny użytek nazywam "pokojowcami", PHANTOM CLUB dostarczy Ci niewielu emocji. Jedynym jej wyróżnikiem jest ogromna - sięgająca 550 - liczba pomieszczeń, które musisz przemierzyć. W każdym z nich na pnącego się po 10 szczeblach drabiny awansu społecznego supermana czekają żarłoczne kwiatki, wysuwające się znienacka spod podłogi obeliski, lewitujące posążki Buddy, tajemniczy mnisi, ohydne mechaniczne pająki i inne przykre na ogół niespodzianki. Co najważniejsze, nie bardzo wiadomo o co, poza bieganiem, przeskakiwaniem przeszkód i zamrażaniem swym oddechem wrogów, chodzi. Bardziej szczegółowe informacje o pełnionej misji uzyskuje się po znalezieniu w jednym z pomieszczeń ekranu i zamrożeniu go. Aby do reszty obrzydzić grającemu życie, przewidziano zgoła odmienne reakcje supermana po powtórnym zetknięciu się z tymi samymi przedmiotami. Kiedy pierwszy raz dotkniesz ruchomej kuli - uzyskasz dodatkowe życie, a ruchomy diament przyspieszy Twoje ruchy. Co będzie za drugim razem - przekonaj się sam. Do wirującej kuli strzelaj bez zastanowienia - uzyskasz wiele cennych punktów. Kiedy w lewym górnym rogu ekranu pojawia się głowa orła - musisz zniszczyć zamieszkującego ten pokój stwora przed jego opuszczeniem.



W sumie - taka sobie grafika, jeszcze gorszy dźwięk i słabiutka fabuła. Kto lubi błądzić - w drogę. Sterujemy joystickiem lub klawiszami kursora i spacją (skoki). Mrożący krew (przeciwników) w żyłach oddech wyzwala przycisk "Fire", "Enter" lub "Copy". "Caps Lock" umożliwia pauzowanie. Kiedy będziesz miał dość - naciśnij "Escape".

Marek Car



PĘTLICZEK - bo pętla jest podstawą programowania. Tu znajdziesz kolejną porcję zadań naszego Klubu Mistrzów Komputera. MĘTLICZEK - bo znajdziesz tu różne różności, związane z minikomputerem tak cienką nitką, że Redakcja już nie bierze za nią odpowiedzialności. Redakcja strony klubowej: Leszek Rudak (ASCII 01), Adam Nowicki (ASCII 02).

KOMUNIKAT KLUBOWY

Rozpoczynamy kolejne półrocze. Dziś prezentujemy aktualną listę członków KMK - aktualną to znaczy na ok. 2 miesiące przed ukazaniem się tego numeru w kioskach. Wszystkie nadesłane do tego czasu rozwiązania zostały sprawdzone, a wyniki wprowadzone do klubowej bazy danych. Lista jest wydrukiem z tej bazy.

W tym miejscu warto dodać jedną uwagę. Regularnie otrzymujemy rozwiązania od wielu osób. Często są to rozwiązania kilku zadań, ale najczęściej dostajemy listy z rozwiązaniem jednego zadania. Dla nas oznacza to, że nowy system ustalania członkostwa KMK spodobał się Czytelnikom. Niżej powtarzamy regulamin Klubu.

Ponawiam apel do Czytelników, Członków i Kandydatów KMK o przysyłanie ciekawostek, krótkich, ale efektownych programików i rozwiązań problemów, o których nikt inny nie pomyślał.

L.R.

REGULAMIN KLUBU MISTRZÓW KOMPUTERA

1. Członkiem Klubu Mistrzów Komputera zostanie Czytelnik, który:

a) przyśle poprawne rozwiązania sześciu dowolnie wybranych zadań. Rozwiązanie każdego z tych zadań nie powinno nadejść do redakcji później niż trzy miesiące od daty ukazania się "Komputera" zawierającego dane zadanie.

b) przyśle dwa oryginalne ułożone przez siebie zadania. Łączny czas na spełnienie wymagań nie jest ograniczony.

2. Członek Klubu Mistrzów Komputera zachowuje członkostwo na następne pół roku, jeżeli w poprzednim półroczu zdobędzie co najmniej 3 punkty. Każde rozwiązanie zadania klubowego daje jeden punkt. Po jednym punkcie zdobywa się także za przysłanie propozycji nowego zadania i za przysłanie czegoś, co może zostać wykorzystane na naszej klubowej stronie.

ZADANIA KLUBOWE

19/88. Szalony Majster buduje konstrukcję z żelbetu. Potrzebne są odcinki prętów zbrojeniowych o długości 240, 600 i 750 centymetrów. Huta dostarcza tylko pręty 12 - metrowe. Proponuję napisać program znajdujący taki sposób cięcia prętów, by ilość traconej stali była jak najmniejsza. Liczby potrzebnych odcinków o poszczególnych długościach powinny być podawane jako zmienne.

20/88. Proponuję napisać program, który przetwarza zadany ciąg znaków w taki sposób, że usuwa z niego fragmenty zgodne z podanym wzorcem.

21/88. Proponuję napisać program wykonujący działania dodawania, odejmowania, mnożenia i dzielenia liczb zespolonych, reprezentowanych w postaci wektorowej lub biegunowej.

(Zadania 20 i 21 pochodzą z listy zadań rozwiązywanych przez studentów III. roku Informatyki Politechniki Wrocławskiej)

A.N.

GWARDZISTA

Programy są często przegrywane masowo i upychane na twardym dysku przez różnych "chomików". "Warta" sprawdza, czy aktualna stacja dysków - ta z której uruchomiono program - jest stacją A. Zazwyczaj numer 0 i symbol A nosi stacja dysków elastycznych. Program uruchomiony z dysku twardego (na przykład numer 2 i symbol C) zawiesi się. Zabezpieczenie niezbyt eleganckie - co zrobić, jeśli sami chcemy ofiarować program koledze, który używa dysku twardego? Po prostu skompilować wersję źródłową jeszcze raz zmieniając uprzednio numer stacji w procedurze "warta"...

Procedura "warta" pyta o hasło. W przykładowym programie jest to zaszyfrowane imię "danusia". Nie można uniknąć zaszyfrowania hasła, gdyż program zawarty w pliku "nazwa.COM" łatwo przejrzeć i znaleźć dziwne na pozór słowo - hasło. Sposób szyfrowania jest prosty: od kodu kolejnej litery hasła jest odejmowana podwojona pozycja litery w haśle. To wystarczy, by wścibski użytkownik Nortona Commandera czy XTREE nie zauważył zaszyfrowanego słowa.

Naturalnie nie możemy przekazywać znajomym tekstu źródłowego programu. Musimy go skompilować do postaci "nazwa.COM". W tym celu używamy dyrektywy O (Options) Turbo Pascala, a następnie poddyrektywy C.

Adam Nowicki

Ps. Podobnie działający programik "WARTOWNIK", przeznaczony dla ZX Spectrum, publikowaliśmy w odcinku 4/87 KMK..

```
program gwardzista;
{ (C) 1987 ADAN & Turbo-Pascal
ver. 3.01A }
procedure warta;
type
  rejestry = record
    AX,BX,CX,DX,BP,SI,DI,DS,
    ES,Flagi : integer;
  end;
      {rekord powinien zawierac na-
      zwy wszystkich rejestrow; mo-
      zna tez uzyc 8-bitowych, np.
      AL,AH,BL,BH,CL,CH,DL,DH :
      byte}
const
        haslo : array [1..7] of
                char = 'b]hmi]S';
                {haslem jest 'da-
                nusia' ; aby je
                zmienic, wystarczy
                zmienic dane tylko
                w powyzszej linii}
var
  reg          : rejestry;
naped,len,n    : integer;
dec            : boolean;
lit            : char;
begin {procedura sprawdzajaca, czy
       uzycie programu legalne;
       sprawdza, czy aktualna
       stacja dyskow jest A, na-
       stepnie haslo}
  with reg do
  begin
    AX:=$1900;
              {funkcja 19h MS-DOS}
    MsDos(reg);
              {wywolanie INT 21h}
    naped:=lo(AX);
              {numer stacji aktualnej}
   {lo(AX) = AL ; mlodszy bajt AX}
  end;
  if naped<>0 then halt;
     {aktualny naped nie jest nape-
     dem A}
  len:=length(haslo);
  clrscr;        {czyszczenie ekranu}
  gotoxy(10,10); writeln(^G,'Podaj ha-
                 slo?');
                         (^G - dzwonek}
  gotoxy(10,11); writeln ('[',len,' liter ]');
  repeat
    gotoxy(10,13);
    dec:=true;
    for n:=1 to len do
    begin
      repeat until keypressed;
      read(Kbd,lit); {czytanie znaku
z klawiat.}
      write(lit);
      if lit<>(chr (ord (haslo[n]) +
2*n)) then
        dec:=false;
    end;
    if not dec then delline;
                   {kasowanie linii}
  until dec;
     {az zostanie podane wlasciwe}
  writeln;
end;
begin
  warta;
  writeln('DROGA WOLNA !');
end.
```

HISTORIA KOMPUTERA

Od powstania pierwszego komputera kamieniami milowymi historii maszyn liczących stawały się kolejne udane modele tych maszyn. Pierwsze komputery powstały w USA, ale inne kraje też wzięły udział w wyścigu ku nowoczesności. Do kolejki maszyn amerykańskich ENIAC i EDVAC dołączyły więc między innymi angielskie: EDSAC (Electronic Delayed Storage Automatic) i MADAM (Manchester Automatic Digital Machine), radzieckie: M-20 i BESM.

Wśród konstruktorów nowoczesnych maszyn liczących nie zabrakło też Polaków. Polscy matematycy od dawna doskonale dawali sobie radę z maszynami, czego najlepszym przykładem jest złamanie szyfru Enigmy. W 1949 roku w nowo zorganizowanym Instytucie Matematycznym PAN powstała Grupa Aparatów Matematycznych. Był to zespół młodych matematyków i konstruktorów, który postawił sobie zadanie zbudowania polskiego komputera.

Pierwszy model GAM-1 był tylko laboratoryjną próbą - nigdy nie wykonał żadnych obliczeń użytkowych. Kolejne modele, nazywane podobnie do amerykańskich: EMAL-1 i EMAL-2 (od Elektronowa Maszyna Automatycznie Licząca), też nie wyszły poza ściany laboratorium.

Dopiero w 1958 roku w Zakładzie Aparatów Matematycznych PAN (tak nazwano przekształconą Grupę Aparatów Matematycznych) skonstruowano komputer o dobrych parametrach. Komputer ten nazwano XYZ-1. Maszyna ta potrafiła w ciągu jednej sekundy dodać ok. 1000 liczb, a pomnożyć ok. 350.

Następne modele maszyn typu XYZ też były udane. Najlepszym ówczesnym polskim produktem była jednak następczyni serii XYZ, maszyna typu ZAM. Ta maszyna, mająca parametry porównywalne z najlepszymi maszynami świata, z powodzeniem była wykorzystywana do obliczeń naukowych - budowa reaktora jądrowego - i konstrukcyjnych. Maszynę tę można obejrzeć w Muzeum Techniki w Warszawie.

L.R.

LISTA CZŁONKÓW KMK (16 maja 1988)

NAZWISKO	MIASTO
Krzysztof Bachurski	RYDZYN PODLASKI
Wojciech Białek	KRZESZOWICE k/Krakowa
Dariusz Bogdański	WROCŁAW
Jacek Cenzartowicz	SZCZECIN
Robert Jabłoński	BIELSK PODLASKI
Dariusz Jankowski	SIEDLCE
Kazimierz Korfanty	RZESZOW
Michał Koźlik	KATOWICE
Wojciech Kromer	GDAŃSK OLIWA
Marek Mitros	AUGUSTOW
Adam Nowicki	WROCŁAW
Jan Okrasiński	WARSZAWA
Zbigniew Orecki	SZCZECIN
Jerzy Saternus	TYCHY BOJSZOWY
Krzysztof Syrek	PŁOCK
Rafał Urbańczyk	TARNOWSKIE GORY
Jacek Zapała	RADOM

FORUM

Drodzy Czytelnicy!

"Forum" to rubryka przeznaczona w całości do Waszej dyspozycji. Piszcie do niej nie tylko o swoich osiągnięciach, nadsyłajcie użyteczne programiki czy "sztuczki i chwyty", które odkryliście, ale piszcie także o problemach, na jakie natrafiacie w pracy z mikrokomputerami. Być może ktoś inny je rozwiązał i będzie mógł Wam tą drogą pomóc. My ze swej strony mamy tylko cztery prośby: o zwięzłe formułowanie listów, o zaznaczanie jakiego typu mikrokomputera dotyczy sprawa, o umieszczanie na kopertach dopisku "Forum" oraz o podawanie wewnątrz listu dokładnego adresu nadawcy. Za publikowane w tej rubryce programy i ciekawe "sztuczki i chwyty" przysługuje honorarium, zgodne z obowiązującymi u nas stawkami. Dla przypomnienia podajemy nasz adres:

PMI "Komputer"
ul. Koszykowa 6a
00-564 Warszawa
"Forum"

Dzisiaj prezentujemy: program Cross służący do wyszukiwania rozkazów w programie napisanym w Basicu (Commodore C16), dwie praktyczne porady dla właścicieli Amstrada CPC464, program umożliwiający wydrukowanie całego ekranu (24 linie) ZX Spectrum oraz program do graficznej prezentacji funkcji przestrzennych dla IBM PC.

Wszystkich zainteresowanych serdecznie zapraszamy do współpracy.

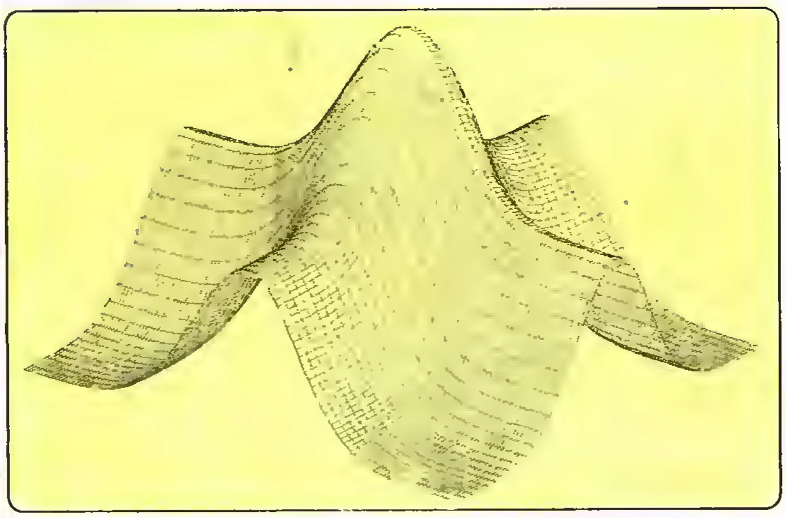
Redakcja

Grafika (IBM PC)

Szanowna Redakcjo!

Przesyłam przykłady graficznego przedstawienia funkcji z=f(x,y). Nie wiem, czy jest to temat ciekawy dla Was, być może ktoś lepiej sobie z tym poradził.

Załączam też główną część programu (Turbo Pascal, IBM PC), która w uproszczeniu tworzy obraz na monitorze. Dla informacji podaję, że procedura obsługi drukarki generuje dane dla niej w ok. 12 minut (IBM AT, koprocesor, 10 MHz).



Proponuję uruchomić program bez zmiany wartości zadanych w deklaracji CONST (wystarczy naciskać Enter).

Następnie można:

- zdefiniować inną funkcję (w FUNCTION z);
- zmieniać granice obszaru, w którym badamy funkcję (xmin, xmax, ymin, ymax, zmin, zmax);
- zmieniać kąt, z jakiego obserwujemy (poziomy i pionowy, oba w granicach 0° - 90°);
- zmieniać zagęszczenie przekrojów funkcji (wielkości podzialx i podzialy).

```
- 1 -

PROGRAM Graph3D;   {W.Panek   1988.02.26}
  CONST podzialX = 64;
        podzialY = 64;
        xmin: Real = -1.6; xmax: Real = 1.6;
        ymin: Real = -1.6; ymax: Real = 1.6;
        zmin: Real = 0.1; zmax: Real = 2;
        fih: Real = 0.5; fiv: Real = 0.1;
        kolor = 1;
  VAR x,y,xleft,xright,ydown,yup,skalaX,skalaY,krokX,krokY,sh,ch,sv,cv: Real;
      pasek,ppasek: ARRAY [0..639] OF Integer;
      start,fin,GraphPen: Integer;

FUNCTION z: Real;
BEGIN z := Exp(-x*x) + Exp(-y*y) END;

FUNCTION TransX(x,y: Real): Integer;
BEGIN TransX := Round((-x*sh+y*ch-xleft)*skalaX) END;

FUNCTION TransY(x,y,z: Real): Integer;
BEGIN TransY := Round((-(x*ch+y*sh)*sv+z*cv-ydown)*skalaY) END;

PROCEDURE Inp(VAR liczba: Real);
  VAR lancuch: STRING[6];
      blad,col,line: Integer;
BEGIN col := WhereX; line := WhereY;
  REPEAT
  GotoXY(col,line); Read(lancuch);
  Val(lancuch,liczba,blad); IF blad > 0 THEN Write(#7)
  UNTIL blad = 0;
GotoXY(col,line); WriteLn(liczba:6:2,'  ')
END;

PROCEDURE Granice;
  VAR tabx,taby,tabz: ARRAY [0..1] OF Real;
      xe,ye: Real;
      i,j,k: Integer;
  PROCEDURE Trans(x,y,z: Real);
  BEGIN xe := - x * sh + y * ch;
        ye := -(x * ch + y * sh) * sv + z * cv
  END;
BEGIN
ClrScr;
Write('xmin = ',xmin:6:2); GotoXY(8,1); Inp(xmin);
Write('xmax = ',xmax:6:2); GotoXY(8,2); Inp(xmax);
Write('ymin = ',ymin:6:2); GotoXY(8,3); Inp(ymin);
Write('ymax = ',ymax:6:2); GotoXY(8,4); Inp(ymax);
Write('zmin = ',zmin:6:2); GotoXY(8,5); Inp(zmin);
Write('zmax = ',zmax:6:2); GotoXY(8,6); Inp(zmax);
WriteLn;
fih := 180 * fih / Pi; fiv := 180 * fiv / Pi;
Write('kat poziomy = ',fih:6:2); GotoXY(15,8); Inp(fih);
Write('kat pionowy = ',fiv:6:2); GotoXY(15,9); Inp(fiv);
fih := fih * Pi / 180; fiv := fiv * Pi / 180;
sh := sin(fih); ch := cos(fih); sv := sin(fiv); cv := cos(fiv);
tabx[0] := xmin; tabx[1] := xmax;
taby[0] := ymin; taby[1] := ymax;
tabz[0] := zmin; tabz[1] := zmax;
Trans(tabx[0],taby[0],tabz[0]);
xleft := xe; xright := xe;
ydown := ye; yup := ye;
  FOR i := 0 TO 1 DO
    FOR j := 0 TO 1 DO
      FOR k := 0 TO 1 DO BEGIN
      Trans(tabx[i],taby[j],tabz[k]);
      IF xe < xleft THEN xleft := xe;
      IF xe > xright THEN xright := xe;
      IF ye < ydown THEN ydown := ye;
      IF ye > yup THEN yup := ye
      END;
skalaX := 639 / (xright - xleft);
skalaY := 199 / (yup - ydown)
END;

PROCEDURE Rob_Pasek;
  VAR i,xe,ye,pxe: Integer;
      zr: Real;
BEGIN
ppasek := pasek;
  FOR i := 0 TO 639 DO pasek[i] := -1;
start := TransX(x,ymin); fin := TransX(x,ymax);
y := ymin; pxe := start;
  REPEAT zr := z;
  IF zr < zmin THEN zr := zmin;
  IF zr > zmax THEN zr := zmax;
  xe := TransX(x,y); ye := TransY(x,y,zr); pasek[xe] := ye;
    IF xe - pxe > 1 THEN
    FOR i := pxe + 1 TO xe - 1 DO
    pasek[i] := Round(pasek[pxe] + (ye - pasek[pxe]) / (xe - pxe) * (i -
pxe));
  y := y + krokY; pxe := xe
  UNTIL y > ymax + 0.4 * krokY
END;

PROCEDURE Rysuj_Pasek;
  VAR i: Integer;
BEGIN
  FOR i := start + 1 TO fin DO BEGIN
    IF ppasek[i] <> -1 THEN BEGIN
    GraphPen := Black;
    IF pasek[i] - ppasek[i] > 1
    THEN Draw(i,200-pasek[i],i,198-ppasek[i],GraphPen);
    IF ppasek[i] - pasek[i] > 1
    THEN Draw(i,198-pasek[i],i,200-ppasek[i],GraphPen);
    GraphPen := kolor
    END;
  Draw(i,199-pasek[i],i-1,199-pasek[i-1],GraphPen)
  END
END;


PROCEDURE Rysuj;
  VAR i,j: Integer;
      kl: Char;
BEGIN
HiRes;
HiResColor(Black);
i := TransX(xmin,ymin); j := TransY(xmin,ymin,zmin);
Draw(i,199-j,TransX(xmax,ymin),199-TransY(xmax,ymin,zmin),GraphPen);
Draw(i,199-j,i,199-TransY(xmin,ymin,zmax),GraphPen);
Draw(i,199-j,TransX(xmin,ymax),199-TransY(xmin,ymax,zmin),GraphPen);
krokX := (xmax - xmin) / podzialX; x := xmin;
krokY := (ymax - ymin) / podzialY;
```

> 30

29 <

```
FOR i := 0 TO 639 DO pasek[i] := -1;
  REPEAT
  IF KeyPressed THEN Exit;
  Rob_Pasek;
  Rysuj_Pasek;
  x := x + krokX
  UNTIL x > xmax + 0.4 * krokX;
i := TransX(xmax,ymax); j := TransY(xmax,ymax,zmin);
Draw(i,199-j,TransX(xmax,ymin),199-TransY(xmax,ymin,zmin),GraphPen);
Draw(i,199-j,TransX(xmin,ymax),199-TransY(xmin,ymax,zmin),GraphPen);
i := TransX(xmin,ymax); j := TransY(xmin,ymax,zmax);
Draw(i,199-j,TransX(xmin,ymin),199-TransY(xmin,ymin,zmax),GraphPen);
Draw(i,199-j,i,199-TransY(xmin,ymax,zmin),GraphPen);
i := TransX(xmax,ymin); j := TransY(xmax,ymin,zmax);
Draw(i,199-j,TransX(xmin,ymin),199-TransY(xmin,ymin,zmax),GraphPen);
Draw(i,199-j,i,199-TransY(xmax,ymin,zmin),GraphPen);
Read(Kbd,k1);
TextMode(C80)
END; {Rysuj}

BEGIN
GraphPen := kolor;
Granice;
Rysuj
END.
```

Program wykorzystuje tryb graficzny o rozdzielczości 640x200 punktów.

Po drobnych przeróbkach można go uruchomić w CPC6128, ale szybkość rysowania drastycznie spada (IBM - 15-30s, Amstrad - kilka minut).

Załączam pozdrowienia

Wacław Panek
Szczecin

Porady (CPC464)

Szanowni Koledzy!

W ramach porad dla użytkowników Amstrada CPC464:

1. Niekiedy trafia nam w ręce zabezpieczony przed wylistowaniem program w Basicu - po rozkazie CAT na ekranie po każdym bloku pojawia się niemiły znak (%). Istnieje dużo programików znoszących zabezpieczenie, wymagają one jednak wgrania do komputera i uruchomienia, co zabiera czas.

Proponuję sposób krótszy, należy wprowadzić w trybie bezpośrednim następujące odbezpieczenie: POKE &AC02,&90:POKE &AC03,&C0:POKE &AC01,&C3 (i Enter).

Teraz możemy załadować nasz zabezpieczony program za pomocą rozkazu LOAD"nazwa" lub LOAD"" i spokojnie wylistować go na ekranie.

2. System operacyjny CPC464 pozwala zapisywać na taśmie magnetofonowej z gęstością 1000 lub 2000 bodów za pomocą rozkazu SPEED WRITE 0 lub SPEED WRITE 1. Przy dużych odstępach międzyblokowych potrzeba wcale niemało taśmy. Taśmę można zaoszczędzić uruchamiając za pomocą rozkazu RUN poniższy programik - uzyskujemy wtedy możliwość zapisu z gęstością 3000 bodów, co nawet pospolita taśma zupełnie dobrze toleruje.

Zapisywany program zrzucamy prostym SAVE (bez określania SPEED WRITE).

```
10 REM ustawienie szybkiego
   zapisu
20 REM na 3000 bodów
30 FOR x=&B700 TO &B709
40 READ v:POKE x,v
50 NEXT
60 CALL &B700
70 DATA &21,&6e,0,&3e,&0f
80 DATA &cd,&68,&bc,&c9,0,0
90 NEW
```

Pozdrowienia

Krzysztof Czerwiński
Kraków

24 Linie (ZX Spectrum)

Przy kopiowaniu zawartości ekranu na drukarkę używanie rozkazu Basica "COPY" nakłada na użytkownika pewne ograniczenia. Na drukarkę przenoszone są 22 linie ekranu, natomiast dwie linie edycyjne są "obcinane". Program "24 LINIE" pozwala ominąć rozkaz COPY i powoduje wydruk 24 linii.

Wywołanie programu następuje przez RANDOMIZE USR 60000.

Po uruchomieniu programu ładującego kod maszynowy nagrywamy na kasetę program w asemblerze za pomocą rozkazu SAVE "24 LINIE" CODE 60000,128

Program nie jest relokowalny.

Ryszard Tegler
Szczecin

```
 10 READ A,N
 20 READ L$: LET L=LEN L$: LET
S=0: LET K=2: LET N=N+10
 30 IF L<=0 THEN STOP
 40 LET Q2=CODE L$(K-1): LET
Q1=CODE L$(K)
 50 LET C=Q1-48-7*(Q1>64)+16*
(Q2-48-7*(Q2>64))
 60 IF K<L THEN POKE A,C: LET
 S=S+C: LET K=K+2: LET A=A+1:
 GO TO 40
 70 IF S-256*INT (S/256)<>C
 THEN PRINT "Blad w linii "
 N: STOP
 80 GO TO 20
 90 REM
100 DATA 60000,100
110 DATA "F306C0210040E5C5CD91"
120 DATA "A8EAC1E1247CE60720E1"
130 DATA "0A7DC6206F3F9FE6F898"
140 DATA "846710E7180DF321001B"
150 DATA "5B0608C5CDA8EAC1105E"
160 DATA "F93E04D3FBFB21005B80"
170 DATA "FD7546AF47772310FC54"
180 DATA "FDCB308E0E21C3D90D5E"
190 DATA "78FE039FE602D3FB5725"
200 DATA "CD541F380A3E04D3FB92"
210 DATA "FBCD93EACF0CDBFB877D"
220 DATA "F830EB0E205E230608D0"
230 DATA "CB12CB13CB1ADBFB1F95"
240 DATA "30FB7AD3FB10F00D20A0"
250 DATA "E9C9B2" ""
```

Cross (Commodore C16)

Szanowna Redakcjo!

Przesyłam swój drugi programik dla C16 *(pierwszy, "Zegar", opublikowaliśmy w "Forum" nr 5/88 - przyp. red.)*. Nazywa się CROSS i służy do wyszukiwania rozkazów w programie napisanym w Basicu. Krótka instrukcja obsługi:

1. wpisujemy program z Monitora i nagrywamy go na taśmie lub dysku;
2. przez "X" wychodzimy z Monitora i wczytujemy program w Basicu;
3. do komórki 1583 wprowadzamy kod odpowiedniego rozkazu;
4. po SYS 1525 komputer wypisuje numery linii, w których ten rozkaz się znajduje.

Poniżej podaję wykaz rozkazów wraz z ich kodami:

128 end	160 close	192 tan	224 char
129 for	161 get	193 atn	225 box
130 next	162 tab(	194 peek	226 circle
131 data	163 new	195 len	227 gshape
132 input#	164 to	195 str$	228 sshape
133 input	165 fn	197 val	229 draw
134 dim	166 spc(	198 asc	230 locate
135 read	167 then	199 chr	231 color
136 let	168 not	200 left$	232 scnlr
137 goto	169 step	201 right$	233 scale
138 run	170 +	202 mid$	234 help
139 if	171 -	203 go	235 do
140 restore	172 *	204 rgr	236 loop
141 gosub	173 /	205 rclr	237 exit
142 return	174 ^	206 rlum	238 directory
143 rem	175 and	207 joy	239 dsave
144 stop	176 or	208 rdot	240 dload
145 on	177 >	209 dec	241 header
146 wait	178 =	210 hex$	242 scratch
147 load	179 <	211 err$	243 collect
148 save	180 sgn	212 instr	244 copy
149 verify	181 int	213 else	245 rename
150 def	182 abs	214 resume	246 backup
151 poke	183 usr	215 trap	247 delete
152 print#	184 fre	216 tron	248 renumber
153 print	185 pos	217 troff	249 key
154 cont	186 sqr	218 sound	250 monitor
155 list	187 rnd	219 vol	251 using
156 clr	188 log	220 auto	252 until
157 cmd	189 exp	221 pudef	253 while
158 sys	190 cos	222 graphic	254 nieużywany
159 open	191 sin	223 paint	255 π

CROSS, to tłumaczony na język maszynowy program z książki F.Schrodera "C16". Dzięki temu, że jest napisany w asemblerze, jest krótszy, szybszy i nie zajmuje pamięci użytkownika.

```
>05f5 a0 00 a9 93 20 d2 ff a5 A5    1138
>05fd 2b 85 d0 a5 2c 85 d1 a5 A5    1100
>0605 d0 18 69 04 85 d4 a5 d1 D1    1060
>060d 69 00 85 d5 b1 d4 aa a0 E0    1234
>0615 22 d0 12 e6 d4 d0 02 e6 E6    1142
>061d d5 b1 d4 aa f0 75 a0 22 22    1387
>0625 f0 56 4c 18 06 e0 00 f0 F0     896
>062d 6a a0 80 d0 4b a0 02 b1 B1    1080
>0635 d0 85 d2 c8 b1 d0 85 d3 D3    1480
>063d a2 05 a9 00 95 d9 ca d0 D0    1112
>0645 fb f8 a0 10 06 d2 26 d3 D3    1140
>064d a2 05 b5 d9 75 d9 c9 10 10    1116
>0655 29 0f 95 d9 ca d0 f3 88 88    1211
>065d d0 ea 18 d8 b5 da 69 30 30    1234
>0665 20 d2 ff e8 a0 05 d0 f4 F4    1410
>066d a9 20 20 d2 ff e8 e0 08 08    1162
>0675 d0 f8 20 e1 ff d0 01 60 60    1273
>067d e6 d4 d0 02 a6 d5 a5 d0 00    1468
>0685 a6 d1 18 69 50 90 01 e8 E8     961
>068d c5 d4 d0 80 a4 d5 f0 03 03    1429
>0695 4c 11 06 b1 d0 aa c8 b1 B1    1031
>069d d0 86 d0 85 d1 88 a5 d0 D0    1401
>06a5 f0 03 4c 04 06 a5 d1 f0 F0     943
>06ad 03 4c 04 06 60 00 00 00 00     185
                                   27593
```

Oto listing programu. Liczby z prawej strony oznaczają sumy kontrolne, które komputer powinien wyświetlić po uruchomieniu programu:

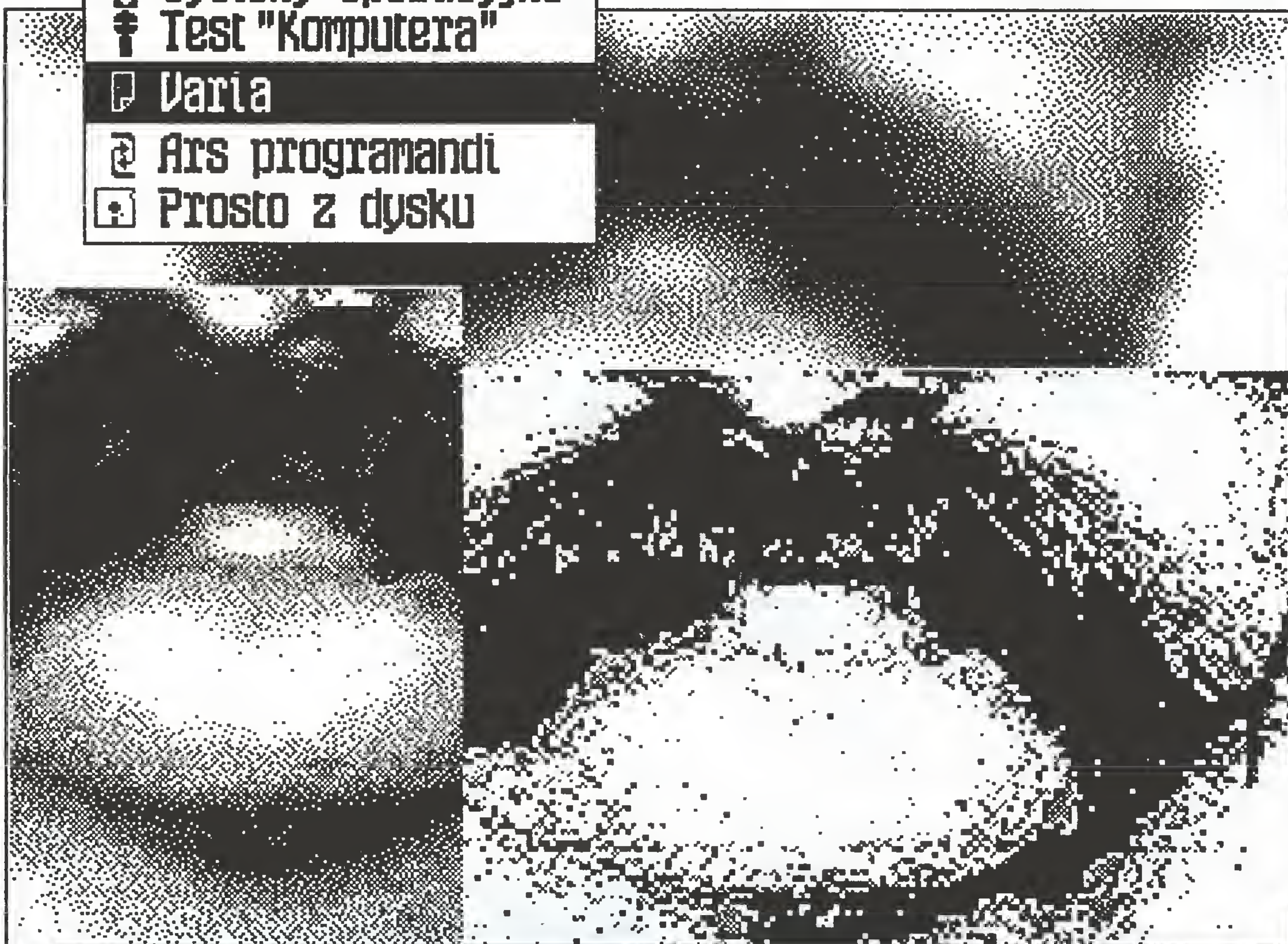
```
10 for x=1525to1709step8:a=0
20 for y=xtox+7:a=a+peek(y)
30 next:printa:s=s+a:next:prints
```

Z poważaniem

Grzegorz Mańturzyk
Radzyń Podlaski



Rys. PIOTR KAKIET



\W pracy\

Władysław Majewski, Jakub Tatarkiewicz, Artur Thielman

Nieśmiertelny robal

Proponujemy Wam twórczą i kształcącą zabawę, w której, podobnie jak w grze w wojny rdzeniowe, do której namawialiśmy Was przed kilkoma miesiącami ("Komputer" 9/87), nie tylko poznajemy cudze światy, ale i tworzymy własne.

Idea "automatu komórkowego", którą chcemy Wam dziś przybliżyć, od dawna nie jest jedynie igraszką - ma ona swe miejsce w dziejach i przyszłości informatyki. Również w fizyce od około trzydziestu lat główną ideą przewodnią jest postulat cechowania lokalnego, zgodnie z którym - w uproszczeniu - cały świat składa się z automacików komórkowych.

Nasze rozważania uzupełniamy przykładowymi programami napisanymi tym razem obok Pascala w Logo - jak się wydaje, język ten jest idealnym narzędziem do generowania pięknych wzorów, będących śladami różnych robali.

Czego uczy Logo?

Czytelnicy "Komputera" znają zapewne ideę Seymoura A.Paperta, twórcy języka Logo: nauka matematyki z pomocą komputera może i powinna być wspaniałą zabawą. Pisząc proste programy w Logo dzieci przy okazji mimowolnie poznają całkiem złożone pojęcia matematyczne! Jest to możliwe m.in. dzięki wykorzystaniu pewnej nietypowej metody definiowania figur geometrycznych. Dla przykładu kwadrat opisujemy nie jako czworobok o równych bokach i kątach, lecz jako zestaw prostych poleceń dla żółwia: "idź prosto k kroków, skręć w prawo o 90 stopni i powtarzaj tę procedurę, aż trafisz do punktu wyjścia".

Prawdziwi matematycy natychmiast zapytają: jak taką definicję rozszerzyć na inne figury? Okazało się, że problem jest nader interesujący i od dawna jest przedmiotem zainteresowania niezwykle ważnego dla rozwoju informatyki działu matematyki zajmującego się automatami komórkowymi.

Cóż to jest automat komórkowy? Jeden z twórców teorii poruszania się po sieciach, John Horton Conway (tak, to ten sam Brytyjczyk, który wymyślił słynną grę LIFE, też będącą zresztą przykładem dwuwymiarowego automatu komórkowego) proponuje rozpocząć poznawanie tego pojęcia od potraktowania całej zabawy jako problemu dla robala (worm). Otóż robale matematyczne są, jak wiadomo, bardzo prymitywne i mają małe móżdżki, tak więc cały ich świat, który mogą przemierzać w poszukiwaniu jedzenia, to linie pewnej zadanej sieci.

> 32

31 <

Z życia robali

W jednym z węzłów tej sieci robal wylęga się z jaja (nie wiemy, kto je tam złożył...) i zaczyna łazić, wyjadając jej poszczególne elementy. Nie może więc przebyć po raz drugi drogi, którą raz już przewędrował - grozi mu to śmiercią głodową.

Robal ma nie tylko mały móżdżek, ale i krótki wzrok, widzi więc jedynie swe najbliższe otoczenie. Może ono obejmować zajmowany węzeł i wychodzące zeń drogi lub także sąsiednie węzły. Tak czy owak nie ma mowy o ogarnięciu myślą całej sieci i zaplanowaniu działań na wiele kroków naprzód.

Biedny robal ma nie tylko mały móżdżek i krótki wzrok, ale i krótką pamięć: nie pamięta przebytej drogi lub - co najwyżej - pamięta tylko ostatni jej zakręt.

Nasz robal nie jest zbyt mądry, ale za to uparty. Natrafiwszy na węzeł wyglądający w określony sposób zawsze wybiera tę samą drogę. Oczywiście różne gatunki robali zachowują się różnie, ale zawsze w sposób dla danego gatunku charakterystyczny. Tylko najwyżej rozwinięte gatunki swe zachowanie uzależniają od ostatniego zakrętu lub stanu sąsiednich węzłów. Inne, jeszcze rzadsze, potrafią ze ściśle określonym prawdopodobieństwem wybrać tę lub inną z dróg.

Często w jednej sieci współżyje wiele, a czasem wręcz wielkie mnóstwo robali, rodzą się nowe (np. w wyniku niektórych spotkań robali jednego lub różnych typów), a nawet po upływie pewnego czasu od przejścia ostatniego robala może odradzać się pożywienie.

Robale pozbawione pożywienia zdychają, choć niektóre z nich potrafią przebyć na głodnego kilka odcinków albo przetrwać krótki przednówek. Samym pożywieniem zadowalają się jednak tylko najprymitywniejsze. Bardziej wysublimowane potrzebują także powietrza oraz właściwego klimatu społecznego i giną w samotności lub w tłoku.

Widzimy już, że opisywany prymitywny model może całkiem dobrze ilustrować zupełnie poważne problemy z zakresu biologii, socjologii (w tłumie zachowujemy się zaskakująco prymitywnie...) czy fizyki (choć interpretacja modelu wymaga dobrej znajomości mechaniki kwantowej).

Istnieją zresztą także prawie dosłowne realizacje tego modelu: paleontolodzy znają ślady pradawnych robali i potrafią rozróżniać ich gatunki na podstawie kształtów rowków pozostawionych w skamieniałym mule.

Automat komórkowy na poważnie

Sądzę, że możemy już teraz (nadal w uproszczeniu!) powiedzieć, że automat komórkowy to n-wymiarowa regularna sieć komórkowa ze skwantowanym czasem, na której zdefiniowano funkcję lub/i obiekty, których wartości lub/i zachowanie w danym punkcie i chwili zależy jedynie od stanu tego punktu i najbliższego otoczenia w poprzedniej chwili.

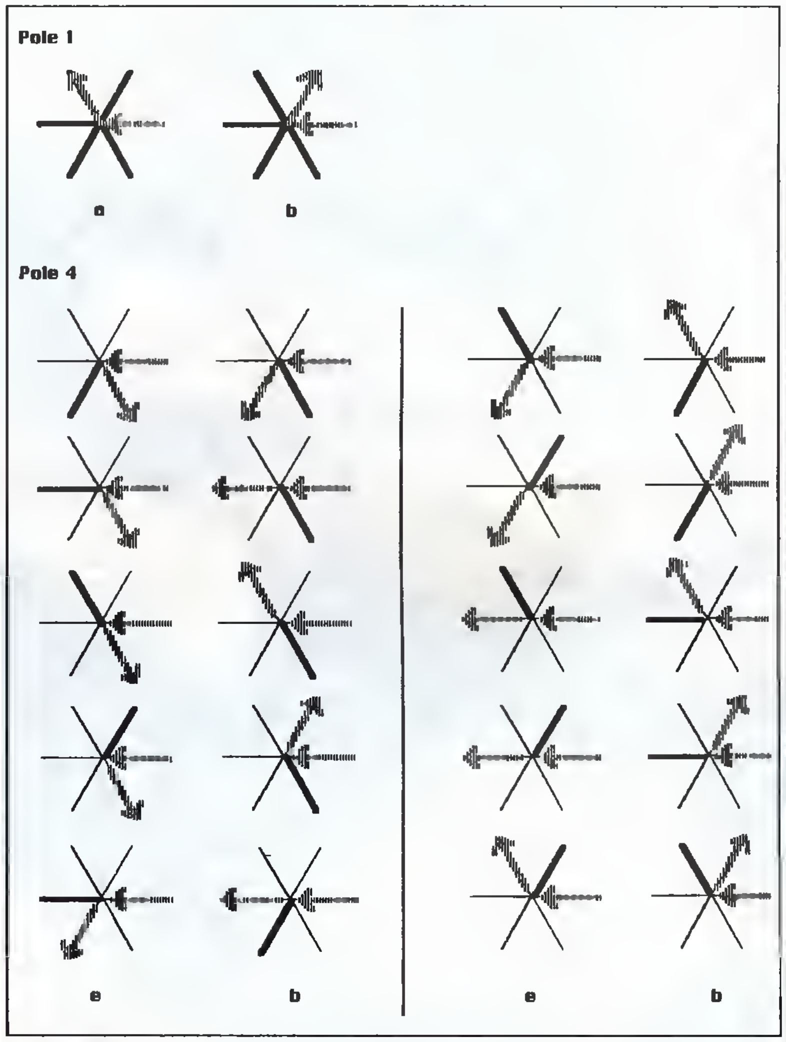
Podstawową cechą automatu komórkowego jest niezależność biegu ewolucji jego niewielkich fragmentów od stanu układu jako całości, a więc brak mechanizmów globalnych lub dalekozasięgowych o natychmiastowym działaniu.

Znaczenie tego modelu wynika nie tylko z istnienia wielu zjawisk, których przebieg można skutecznie analizować za jego pomocą, np. dzięki jego podobieństwu do kwantowej sieci krystalicznej, lecz także z łatwości komputerowej symulacji jego zachowania się i podobieństwa do budowy przyszłych wieloprocesorowych maszyn liczących. Już za parę lat sieci transputerowe, komputery matrycowe i "wielosześcienne", o których pisaliśmy w numerze 3/88, będą mogły analizować z gigantyczną prędkością zachowanie wszystkich modeli, które - jak automaty komórkowe - nie wymagają synchronicznego obliczania globalnych cech modelu.

Automaty i komputery

Nie tylko przyszłość komputerów ściśle wiąże się z automatami komórkowymi, ale i ich historia. Zgodnie z ideą ojca współczesnych maszyn liczących - Johna von Neumanna - komputer składa się z procesora i pamięci, w której mieszczą się zakodowane polecenia dla procesora (mówiąc w skrócie - program) i dane podlegające przekształceniom.

Fizycznie nie ma żadnej różnicy pomiędzy danymi a programem. Są one tak samo kodowane jako ciągi zer i jedynek i tak samo zapisywane jako układy zatkanych i przewodzących tranzystorów. W takt impulsów zegara procesor pobiera zawartość kolejnej komórki pamięci, rozpoznaje (w optymistycznym przypadku) rozkaz i wykonuje go zgodnie ze swym mikroprogramem wpisanym w układ elektryczny.



Celem pracy komputera jest zwykle przejście od jakiegoś stanu zawartości pamięci, zadanego przez programistę, do innego, zawierającego interesujący nas wynik. Może to być równie dobrze poważny, naukowy rachunek, co i gra z piękną grafiką i efektami dźwiękowymi. Na tym najbardziej podstawowym poziomie będzie to jednak to samo - jakieś zmiany stanu pamięci.

Popatrzmy teraz w ten sam sposób na automat komórkowy. I tu też mamy "pamięć" - naszą sieć pól. Tak jak poprzednio zadajemy początkową zawartość pamięci i tak jak od komputera oczekujemy od automatu jej przekształcenia. W drugim z prezentowanych dziś prostych modeli nasz procesor pobiera zawartość trzech komórek i wykonuje rozkaz: zgodnie ze swym mikroprogramem - regułą wpisuje w kolejne miejsca wyniki. Pracując w ten sposób obiega całą "pamięć" i kończy działanie. Możemy więc automat komórkowy uznać za model najprostszego komputera, działającego, mimo małej pamięci, bardzo podobnie do swych troszkę zdolniejszych kolegów.

Robal na sześciokątach i trzy przykazania

Po wykładzie czas na zabawę. Proponujemy Wam bliższe poznanie dwóch rodzin robali, żyjących na płaskiej sieci sześciokątnej oraz na jednowymiarowym odcinku.

Studia nad robalami rozpoczęto przed około 20 laty od dwuwymiarowej sieci kwadratowej (czyli kartki w kratkę zamazywanej podczas nudnych narad), jednakże uzyskiwane na niej wzory nie są najciekawsze - proponujemy samodzielnie zastanowić się dlaczego! Definiowanie natomiast robali na sieci sześciokątnej (dla komputerów jest ona równie dobra) prowadzi do nadzwyczaj ładnych i skomplikowanych tras.

Zabawmy się więc w Pana B. (Pan B. to oczywiście Programista Automatów Naukowo-Badawczych) i zadekretujmy - jeszcze przed urodzeniem się pierwszego naszego robala - zbiór ogólnych reguł poruszania się robali po ich sześciokątnym świecie oraz rozmiary segmentów sieci, tak by robalom pozostało tylko chodzić, chodzić, chodzić...

Utarło się, że Pan B. we wszystkich budowanych dla zabawy sieciach zwykł (bo przecież nikt nie może mu nic nakazać!) stosować trzy podstawowe zasady (ograniczenia wybranego modelu):

1. Jeżeli wszystkie segmenty prowadzące do odwiedzonego przez robala węzła zostały już zjedzone, robal zdycha. W przypadku sieci o parzystej liczbie segmentów w węźle oznacza to, że robal

zdycha po (w sieci sześciokątnej trzykrotnym) powrocie do punktu wyjścia.

2. Jeżeli w węźle pozostał tylko jeden nie zjedzony segment, robal musi po nim pójść.

3. Jeżeli żaden z segmentów w węźle (prócz tego, po którym robal tam dotarł) nie został zjedzony, robal skręca w prawo. Jest to tzw. prawicowe odchylenie robala, będące zapewne wynikiem praworęczności Pana B.

Najdawniejsze legendy głoszą, że był niegdyś pono także tzw. Lewy Pan B., ale tworzył on jedynie dzieła znane z doświadczeń Pana B., tyle że na opak.

Wokół 1296 światów

Robal żyjący w zgodzie z przykazaniami trafiając na węzeł z kilkoma nie wyjedzonymi jeszcze drogami musi wybrać jedną z nich. W węźle bez wyjedzonego segmentu (pole 1 rysunku 1) może skręcić w prawo łagodnie (wybór "a" - jest to tzw. robal łagodny) lub też ostro (wybór "b" - tzw. robal ostry).

Napotykając węzeł z jednym wyjedzonym segmentem, a więc po raz pierwszy wracając do punktu wyjścia (pole 2 rysunku 1) robal wybiera jedną z czterech nietkniętych dróg, mamy więc robale typu a,b,c oraz d (zakładamy, że postępowanie robali nie zależy od wzajemnego położenia segmentu wyjedzonego dawniej i przed chwilą. Rozróżnienie takie nie dałoby jednak nowych wzorów - zastanówcie się dlaczego?)

Węzły z dwoma wyjedzonymi segmentami (oprócz segmentu, po którym robal się zbliża) mogą mieć dla robala dziesięć możliwych kształtów, a w każdym z nich robal może wybrać jedną z trzech dróg, co oznaczałoby 3 do dziesiątej, czyli 59.049 ras robali. Robal łagodny może jednak napotkać tylko węzeł z wyjedzonymi segmentami oddalonymi o 120 stopni, a robal ostry - o 60 stopni. Każdy z nich może więc natrafić na cztery różne sytuacje, a w każdej z nich zachować się na trzy sposoby. Łącznie mamy tu 81 odmian robaków, oznaczanych od aaaa przez acac do cccc. Nie podajemy rysunków, ale odpowiednie układy łatwo sobie można wyobrazić. Szczególnie zainteresowanym polecamy artykuł Martina Gardnera w "Scientific American" z listopada 1973 r., str. 120.

Pozostaje jeszcze pole 4 rysunku 1., na którym robalowi pozostają tylko dwie wolne drogi - znów może powstać tu 10 sytuacji, ale tylko dwa istotnie różne style zachowania się robala.

Węzeł z czterema lub pięcioma wyjedzonymi segmentami nie daje robalowi wyboru: może się oddalić jedyną pozostałą drogą lub pożegnać z tym najlepszym ze światów.

W sumie na sieci sześciokątnej w zgodzie z przykazaniami żyć może 2*4*81*2 = 1296 odmian krótkowzrocznych i nic nie pamiętających robali.

Do ich rozróżniania używa się notacji (nazwisk robali) o brzmieniu np. 1a2b3acac4a, co znaczy, że napotykając pole typu 1 robal wybiera zawsze drogę a itd. Notacja ta pozwala więc opisać robala za pomocą jedenastu symboli.

Robale na start!

Badania komputerowe wykazały, że 209 reguł generuje unikalne wzory, tzn. różne od tworzonych przez wszelkie inne robale, 46 dróg jest produkowanych przez dwa robale, 44 zaś przez trzy lub więcej. Ostatecznie grupa prostych robali daje 299 różnych wzorów. Napisaliśmy "prostych", gdyż matematycy zaczęli oczywiście natychmiast rozważać robale uogólnione w zasugerowany wcześniej sposób lub np. takie, które mogą poruszać się do przodu o zmienną liczbę segmentów w linii prostej. Ale to już dla nas zbyt skomplikowane, tym bardziej, że do dziś nie wszystkie drogi robali prostych przebadano do końca!

Czytelnicy, którzy uruchomili załączony prosty program w Logo, wiedzą, że wiele robali umiera bardzo szybko, inne, np. 1a2d3cbaa4b dają piękne, symetryczne wzory. Pewne robale są zaś nieśmiertelne - taki niepozorny 1a2d3caaa4b będzie sobie żył a żył, choć nigdy już nie powróci do stron ojczystych. Jeszcze inne robale nawijają się w nieskończoność wokół miejsca narodzenia. Najdłuższy śmiertelny robal nazywa się 1a2d3cbac4b i trwa przez... 220142 segmenty, co prawdopodobnie oznacza dla typowego domowego komputerka niemalże nieskończoność.

Na tym jednak nie koniec: istnieje parę robali, o których dotąd nie wiadomo, czy są periodyczne (w sensie tworzenia symetrycznego wzoru), czy nieskończone, a może jednak skończone? Po prostu studia komputerowe aż do wielu milionów zjedzonych segmentów nie dały odpowiedzi wprost. Napotyka się tu zresztą pewien problem praktyczny: jak ustalić, czy dany wzór jest symetryczny, jeśli jego wymiary wiele tysięcy razy przewyższają wymiary nie tylko ekranu komputera, ale i typowego papieru do plotera?!!! Taki 1b2a3bcaa4b znany jest z życia przez co najmniej 10 milionów segmentów i ciągle o nim nic nie wiemy. Być może zastosowanie szybszych komputerów pozwoli rozstrzygnąć ten problem.

Ambitnych Czytelników zachęcamy do własnych studiów nad robalami prostymi. Oczekujemy nadesłania nam najładniejszych robali (tylko nazw lub, lepiej, wydruków). Postaramy się je zaprezentować w jednym z kolejnych numerów.

Namawiamy także do odgadnięcia nazwisk prezentowanych dziś przykładowych robali.

Program dla robala

Załączony program w Logo nie pozwala śledzić losów robala zbyt długo - jedynie do momentu, gdy opuści on ekran, ale ma kapitalną zaletę: żółw działa jak... symulowany robal. Poruszając się po ekranie sprawdza on stan otoczenia węzła, do którego trafił, po czym podejmuje decyzję, co czynić dalej.

Z informatycznego punktu widzenia nie jest to rozwiązanie najszybsze ani najelegantsze, natomiast dobrze oddaje cechy symetrii problemu i jego prostotę. Każdy inny wariant, zakładający osobne pamiętanie np. tablicy stanów poszczególnych dróg, musiałby być sztuczny i pamięciochłonny.

Najprostszy robal

Zajmijmy się teraz rodziną prostych automatów komórkowych, których przestrzeń życiowa obejmuje zaledwie kilkadziesiąt punktów.

Poznawanie ich proponujemy rozpocząć od uruchomienia podanego obok programu Automacik. Kto to uczynił, ten widział nasz automat w akcji. "Żyje" on w świecie złożonym z 80 liniowo uporządkowanych punktów, przy czym, by uniknąć kłopotów z pierwszą i ostatnią komórką, sklejamy nasz odcinek w kółko. Każdy z punktów "świata" może być "wolny" lub "zajęty".

Krok czasowy automatu polega na sprawdzeniu zawartości trzech sąsiednich pól w układzie początkowym, a następnie ustaleniu zgodnie z zadaną regułą stanu środkowego pola w układzie końcowym.

Oznaczmy teraz przez **i** numer pola, jego początkowy stan przez **xi**, końcowy zaś przez **yi**. Efekt wykonania kroku można zapisać w postaci

$$yi = r(xi-1, xi, xi+1),$$

gdzie r oznacza regułę. Zestaw argumentów xi-1, xi, xi+1 może przyjmować 8 różnych wartości, istnieje więc 2 do ósmej, czyli 256 różnych reguł. Oto jak są one numerowane - na przykładzie, skądinąd bardzo ciekawej, reguły 90:

(xi-1, xi, xi+1)	111	110	101	100	011	010	001	000
yi	0	1	0	1	1	0	1	0

Zera i jedynki pod kreskami to dwójkowa postać liczby 90.

Mamy więc 256 reguł i możemy się zabawić np. w ich klasyfikację. W tym celu warto nieco poeksperymentować. Zbadajmy na początek, jak wygląda ewolucja Automaciku od układu początkowego, wybranego przypadkowo (komputer "losuje" stan kolejnej komórki) i z układu początkowego składającego się z jednej lub kilku komórek położonych na niewielkim odcinku.

Szczególnie ciekawie zachowuje się Automacik z regułą 90. Ma on wyjątkową właściwość. Stan w chwili będącej całkowitą potęgą dwójki jest po prostu "zdwojonym" stanem początkowym, a i w chwilach pośrednich można te "zwielokrotnienia" odnaleźć.

Klasyfikacja

Oglądając wyniki, jakie daje stosowanie różnych reguł, zauważymy po pewnym czasie, że dają się one podzielić na trzy rodzaje. Po pierwsze - układ szybko ginie. Ma to na przykład miejsce dla reguły 0 (odpowiedź na każdy układ początkowy jest pusta), ale także 8, 40, 96, 168. Po drugie - i tak jest najczęściej, bo aż w trzech czwartych wszystkich reguł - od pewnego miejsca układ powtarza się co kilka (zwykle jeden do ośmiu) kroków, np. dla reguł 2, 34, 74, 154. Po trzecie wreszcie, początkowy układ zawarty w niewielkim obszarze rozszerza się aż do wypełnienia całego świata, widać wiele podobnych trójek, np. reguły 18, 54, 90, 182.

Poświęćmy trochę uwagi temu podziałowi, także dlatego, że pozwala wybrać sobie reguły "ciekawsze" i "mniej ciekawe" wizualnie. Zauważyliście już pewnie, że gdy reguła jest zadana liczbą nieparzystą, to mały układ początkowy już w następnym kroku zapełnia większość "świata". Dzieje się tak dlatego, że liczba nieparzysta w postaci dwójkowej ma jedynkę na ostatnim miejscu. Uważny Czytelnik dostrzeże, że jedynka ta jest wynikiem, gdy wszystkie trzy badane pola są wolne, co ma miejsce w całej pustej "reszcie

> 34

33 <

świata" w naszym układzie początkowym. Jak więc poprawnie zaklasyfikować np. regułę 235 czy 7? Odpowiedzi poszukajmy w programie Automacik2.

Ten program obsługuje naprawdę dwa niezależne C.A. z tą samą regułą. Losuje dla obydwu ten sam losowy układ początkowy, następnie w jednym z automatów dokonuje zmiany w niewielkim odcinku. Na ekranie kropki oznaczają komórki zajęte w obydwu automatach, zaś zapełnione pola - komórki różniące te automaty (wolne w jednym a zajęte w drugim). Nie widać już tu różnic między liczbami parzystymi a nieparzystymi. Bez zastanowienia wiemy, do której z trzech klas zaliczyć oglądaną regułę. Możemy mieć jeszcze wątpliwości co do kilku C.A., które jak 14 czy 25 bardzo długo "biorą rozbieg" zanim staną się okresowe. Na niewielkim "świecie" mogą nawet po prostu nie zdążyć przed wpadnięciem w okresowość z innego powodu, o którym opowiem za chwilę. Tymczasem jeszcze zajmijmy się zabawą. Otóż nie mam wątpliwości, że obserwując w pierwszym czy drugim programie działanie C.A. z regułami 10, 80, 175, 245 stwierdzicie, że jest ono bardzo podobne. To samo dotyczy np. czwórki 28, 70, 157, 199. Chcę wyjaśnić dlaczego. Zauważmy, że wydają nam się podobne obrazki, w których zastąpimy komórki wolne przez zajęte (i odwrotnie). Pamiętajmy jednak zrobić to zarówno w układzie początkowym jak i w wynikach (odpowiednio nad kreską i pod kreską). Np. reguła 30

111	110	101	100	011	010	001	000
0	0	0	1	1	1	1	0

przejdzie w ten sposób na

111	110	101	100	011	010	001	000
1	0	0	0	0	1	1	1

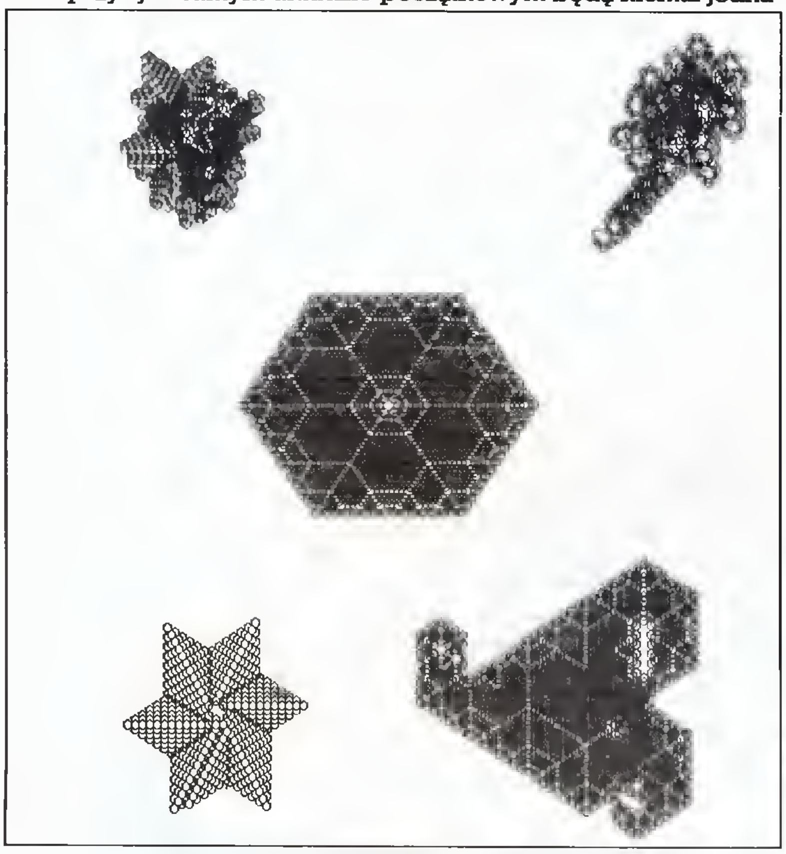
czyli 135. Odpowiada to zamianie w liczbie binarnej, określającej regułę wszystkich zer na jedynki i odwrotnie, oraz zapisaniu jej "od końca". Taką zmianę reguły będę nazywał "zaprzeczeniem". Drugi rodzaj podobieństwa obrazków obserwujemy, gdy automat tylko "patrzy z drugiej strony", czyli gdy przestawimy wyniki odpowiadające (110) i (011) oraz (100) i (001). Reguła 30 przejdzie teraz na

111	110	101	100	011	010	001	000
0	1	0	1	0	1	1	0

czyli 86. Tę zmianę nazywam "odbiciem". Z kolei "zaprzeczenie" zastosowane do liczby 86 da

111	110	101	100	011	010	001	000
1	0	0	1	0	1	0	1

czyli 149. Obrazki "rysowane" przez automaty z regułami 30, 86, 135 i 149 przy tym samym układzie początkowym będą niemal jednakowe. Tak postępując możemy tworzyć dalsze czwórki (lub pary, albowiem zdarza się, że procedura "odbicia" nic nie zmienia, np. dla liczby 22, albo daje to samo co "zaprzeczenie", np. dla liczby 29). Zgadzam się, że wygląda to na nieciekawą matematyczną udrękę z zer i jedynek. Temu, kto zechce obejrzeć wiele różnych prostych automatów, oszczędza jednak pracy - wystarczy zbadać jeden C.A. by znać zachowanie całej czwórki (a co najmniej pary).



Teraz wyjawię powód, dla którego wszystkie oglądane dotąd automaty po pewnym czasie dają powtarzające się okresowo wyniki. Dzieje się tak dlatego, że "świat" naszego C.A. jest skończony. Na skończonej liczbie (powiedzmy n) pól można zadać tylko skończoną liczbę różnych układów wolnych i zajętych komórek - istnieje skończenie wiele n-cyfrowych liczb dwójkowych (konkretnie 2n). Zwykle większość z tych układów nie może być efektem ewolucji C.A. od jakiegoś innego układu, co jeszcze zmniejsza liczbę kroków, w których ewolucja może dawać różne wyniki. Jeżeli zaś jeden układ się powtórzy, to i wszystkie po nim następujące, bo ta sama reguła zastosowana dwa razy do tego samego układu nie może dać dwu różnych wyników! Związany z tym jest jeszcze jeden fakt. Ewolucji C.A. zwykle nie da się odwrócić, czyli znając jakiś układ nie możemy stwierdzić, jaki był poprzedni (co najwyżej może udać nam się dowieść, że go nie było). Wśród 256 prostych C.A. tylko dla 32 (może "aż 32", bo w bardziej złożonych C.A. jest ich jeszcze mniej) da się podać receptę na "chodzenie do tyłu". Ciekawe, czy komuś uda się podać przykład takiej reguły różny od 204 (przekształcenie tożsamościowe - nic w układzie nie zmienia), 170 i 240 (przesunięcia o 1 w lewo i w prawo) i ich "zaprzeczeń".

O automatach prostych niewiele więcej da się powiedzieć. Odrzućmy więc ich ograniczenia i zobaczmy coś więcej - może znajdziemy te same własności i w innych automatach, może uda nam się zaobserwować coś nowego?

```
program Automacik (input, output);
                                   {prosty automat komórkowy}
const
     Szerokość_Ekranu = 80;
     Szerokość_Pola = 80;
     Pusta = 0;
     Pełna = 1;
type
     Słownik = Pusta..Pełna;
     Plansza = array [1..Szerokość_Pola] of Słownik;
var
     Pole_1 : Plansza;      {Stan początkowy}
     Pole_2 : Plansza;      {Stan końcowy}
     Reguła : array [0..7] of Słownik;
     i : integer;           {Indeks pętli}
     h : char;              {Znak wejściowy}
procedure Init ;             {Przygotowuje automat do startu}
       procedure Czytaj_Regułę;
                        {Czyta regułę w postaci dziesiętnej}
       var
            ile : 0..255;
            begin    Czytaj_Regułę
            writeln ;
            write('Podaj regułę [0..255] : ');
            readln(ile);
            writeln('Przyjmuję regułę ',ile);
            for i := 0 to 7 do begin
               Reguła[i] := ile mod 2;
               ile := ile div 2;
            end;         {for}
            writeln ;
       end;              {Czytaj_regułę}
          procedure Czytaj_Stan ;
          {Czyta stan z klawiatury}
                var
                      Linia : string[Szerokosc_Pola] ;
                      {Linia wejsciowa}
                      len : integer; {Dlugosc Linii}
                      ile : integer; {Zmienna pomocnicza}
             begin {Czytaj_Stan}
                writeln('Wpisz stan poczatkowy :');
                i := 1;
                readln(Linia );
                len := Length(Linia );
                ile := (Szerokość_Pola - len) div 2;
                for i := 1 to ile do begin
                      Pole_1[i] := Pusta;
                   end;
                for i := ile + 1 to ile + len do begin
                      if Linia[i - ile] = ' '
```

```
                  then begin
                        Pole_1[i] := Pusta;
                  end
                  else begin
                        Pole_1[i] := Pelna;
                  end;
            end; {for}
            for i := ile + len + 1 to Szerokość_Pola do begin
                  Pole_1[i] := Pusta;
            end;
      end; czytaj_stan)
      procedure Losuj_Stan ;
      Wyznacza stan początkowy pseudolosowo
        begin
          for i := 1 to Szerokość_Pola do begin
                Pole_1[i] := Random(2);
          end;
        end;
  begin      Init
      Czytaj_Regułę ;
      write('Stan początkowy zadany czy losowy [Z/L] ? ');
      readln(h);
      if (h = 'Z') or (h = 'z') then begin
            Czytaj_Stan ;
        end
       else begin
            Losuj_Stan ;
        end;
  end;
procedure Krok_Czasowy ; Wykonuje krok
      function Numer : integer;
      {oblicza początkowy stan otoczenia danej komórki}
            var
                  ile : integer; {Zmienna pomocnicza}
                  Otoczenie : array [1..3] of Slownik;
                  j : integer; {Indeks petli}
            procedure WybierzOtoczenie ;
            {określa otoczenie komórki}
              begin
                  if i = 1 then begin
                    Otoczenie[1]      :=      Pole_1[Szero-
                    kość_Pola];
                    end
                  else begin
                        Otoczenie[1] := Pole_1[i - 1];
                    end;
                  Otoczenie[2] := Pole_1[i];
                  if i = Szerokość_Pola then begin
                        Otoczenie[3] := Pole_1[1];
                    end
                  else begin
                        Otoczenie[3] := Pole_1[i + 1];
                    end;
              end;
                  begin Numer
            Wybierz_Otoczenie ;
            ile := 0;
            for j := 1 to 3 do begin
                  ile := ile*2 + Otoczenie[j];
              end;
            Numer := ile;
        end;
  begin      Krok_Czasowy
      for i := 1 to Szerokość_Pola do begin
            Pole_2[i] := Regula[Numer ];
        end;
      Pole_1 := Pole_2;
    end;
procedure Wydruk ;
{Wydruk jednej linii na ekranie}
    begin
      for i := 1 to Szerokość_Pola do begin
            if Pole_1[i] = Pusta then begin
                  write(' ');
              end
            else begin
                  write(chr(3));
              end;
        end;
      if Szerokość_Pola < Szerokość_Ekranu then begin
            writeln ;
        end;
        end;
begin     main
        Init ;
        repeat
              if Keypressed then begin
                    Wydruk ;
                    Krok_Czasowy ;
              end;
        until false ;
end.
```

```
program Dwa_Automaty (input, output);
                  {bada różnice między dwoma automatami}
const [...]       {stałe identyczne, jak w programie Automacik}
type [...]        {typy identyczne, jak w programie Automacik}
var [...]         {zmienne globalne identyczne, jak w programie
                  Automacik, plus jedna nowa:}
            Pole_3 : Plansza; {Drugi stan pocz.}
procedure Init ;
                  {Przygotowuje automat do startu}
            procedure Czytaj_regułę ;
          [...]{identyczna jak w programie Automacik}
            procedure Losuj_Stan ;
          [...]{identyczna jak w programie Automacik}
            procedure Czytaj_Stan ;
            {Czyta roznice z klawiatury}
                  var
                        Linia : string[Szerokosc_Pola] ;
                        {Linia wejsciowa}
                        len : integer; {Dlugosc Linii}
                        ile : integer; {Zmienna pomocnicza}
                  begin       {Czytaj_Stan}
                        writeln('Wpisz roznice :');
                        i := 1;
                        readln(Linia );
                        len := Length(Linia );
                        ile := (Szerokosc_Pola - len) div 2;
                        for i := 1 to ile do begin
                              Pole_3[i] := Pole_1[i];
                          end;
                        for i := ile + 1 to ile + len do begin
                              if Linia[i - ile] = ' ' then begin
                                    Pole_3[i] := Pole_1[i];
                                end
                              else begin
                                    Pole_3[i] := 1-Pole_1[i];
                                end;
                          end;
                        for i := ile + len + 1 to Szerokosc_Pola do begin
                              Pole_3[i] := Pole_1[i];
                          end;
                  end;
        begin       Init
              Czytaj_Regule ;
              Losuj_Stan ;
              Czytaj_Stan ;
        end;
    procedure Krok_Czasowy(var Pole : Plansza);
    {Wykonuje jeden krok czasowy}
              function Numer : integer;
              {Wylicza numer wyniku dla danej komorki}
                  {funkcja identyczna jak w programie Automacik}
        begin       Krok_Czasowy
              for i := 1 to Szerokosc_Pola do begin
                    Pole_2[i] := Regula[Numer ];
                end;
              Pole := Pole_2;
        end;
    procedure Wydruk ;
    {Wydruk jednej linii na ekranie}
[...]   {identyczna jak w programie Automacik}
    begin {main - dwa_automaty}
        Init ;
        repeat
              if Keypressed then begin
                    Wydruk ;
                    Krok_Czasowy(Pole_1);
                    Krok_Czasowy(Pole_3);
                end;
        until false ;
end.
```

\W pracy\

Andrzej Kadlof

Zabawy z ekranem

Każdy wie, że porządna kopia komputera IBM PC ma 640 KB (kilobajtów) pamięci. Tyle "widzi" system operacyjny MS-DOS i tyle można wykorzystywać we własnych programach. Nie jest to jednak cała pamięć dostępna w tym komputerze. Zainstalowane karty sterowników wyświetlacza ekranu dysponują dodatkową pamięcią, w której przechowują dane o stanie ekranu. Zależnie od zainstalowanej karty może to być od 16 kB dla karty CGA do 256 kB dla kart EGA. W pewnych trybach pracy tylko część tej pamięci jest wykorzystywana, a pozostałą można spożytkować do własnych celów.

Warto od razu zaznaczyć, że bezpośrednie korzystanie z własności sprzętowych komputera, z pominięciem systemu operacyjnego, nie jest zazwyczaj wskazane. Głównym argumentem jest ryzyko, że taki program nie zechce poprawnie działać na innych modelach i ewentualnie będzie wymagał pracochłonnych adaptacji (na pewno pojawią się kłopoty, gdy program będzie musiał działać z wszystkimi kartami wyświetlaczy ekranowych). Co gorsza, tego typu programy mogą stwarzać problemy pracując z różnymi systemami okien lub w sieciach. Bezpośredni dostęp do pamięci ekranu w komputerach klasy IBM PC stał się jednak zjawiskiem tak powszechnym, że w razie ewentualnych tarapatów wpadniemy w nie, w wyśmienitym towarzystwie najlepszych światowych producentów oprogramowania (najdalej poszła chyba firma Borland, programy pisane w Turbo Pascalu 4.0 niemal standardowo omijają pośrednictwo DOS-u i BIOS-u przy obsłudze ekranu). W zamian od zaraz zyskamy pewne możliwości nieosiągalne innymi sposobami.

Postępowanie takie ma bowiem cały szereg zalet. Przede wszystkim pozwala na błyskawiczne zmiany ekranu i to bez utraty jego zawartości. Gotowe strony tekstu z instrukcjami czy wskazówkami dla użytkownika można przechowywać na dysku; ich wywołanie na ekran wymaga kilku bajtów kodu i wcale nie musi zajmować obszaru danych programu. W wersjach Turbo Pascala poniżej 4.0 jest to dość istotne. Giną wreszcie problemy z wypełnianiem dwudziestego piątego wiersza ekranu, bez powodowania automatycznego jego przesunięcia o jedną linię w górę. "Legalnymi" sposobami z poziomu Turbo Pascala (bez odwołań do funkcji BIOS-u) nie potrafię tego osiągnąć.

Poniższe przykłady napisane zostały w Turbo Pascalu, ale wykorzystane w nich techniki mogą być stosowane w innych językach, jeśli istnieje w nich jakiś mechanizm bezpośredniego dostępu do pamięci.

Będę zakładać, że w programie zostały umieszczone deklaracje:

```
const adres_ekranu  = $B000;
      adres_linii   = $F00;              {dziesiętnie 3860}
type symbol = record
                znak    : char;
                atrybut : byte
              end
     obraz  = array[1..2000] of symbol;
     wiersz = string[80];
var ekran : obraz absolute adres_ekranu: $0000;
    linia : array[1..80] of symbol
                 absolute adres_ekranu: adres_linii;
```

Stała **adres_ekranu**, to adres segmentu zawierającego pamięć ekranu na karcie Hercules. Dla karty CGA należy wstawić **$B800**. Druga stała **adres_linii**, to adres 25 wiersza na ekranie.

Pierwszy przykład demonstruje szybkość zapełnienia ekranu. Porównajcie czas działania poniższych dwóch wariantów zapełnienia ekranu tym samym znakiem:

```
for i := 1 to 2000 do write('X');
```
oraz
```
for i := 1 to 2000 do ekran[i].znak := 'X';
```

Różnica jest wręcz niewiarygodna! Zwróćcie ponadto uwagę na to, że w drugiej metodzie zapełnił się cały ekran i jego zawartość nie przesunęła się do góry. Jest to szczególnie cenne, gdy chcemy na przykład cały ekran zamknąć w ramce. Bez grzebania w pamięci ekranu musielibyśmy ograniczyć się do górnych 24 wierszy.

Drugi przykład pokazuje wykorzystanie tych metod do przywoływania na monitor ekranów ze wskazówkami.

Przypuśćmy, że mamy przygotowany oddzielnie na dysku zbiór z gotowymi podpowiedziami utworzony z oddzielnych ekranów. Może to być plik typu sufler:

```
type sufler = file of obraz;
```
Poniższa procedura pozwala wczytać dany ekran bezpośrednio na monitor:
```
procedure podpowiedz ( nr_ekranu : integer );
  var f : sufler;
      zapasowy : ekran;
begin
assign(f,'ratuj.suf');
reset(f);
Move(ekran,zapasowy,4000);
Seek(nr_ekranu);
Read(f,ekran);
Close(f);
repeat until KeyPressed;
Move(zapasowy,ekran)
end; {podpowiedz}
```

Nie tylko zyskujemy na czasie zapełniania ekranu, ale również nie zaśmiecamy kodu programu stałymi i nie zawsze potrzebnymi tekstami. Znacznie upraszcza się też modyfikacja tych tekstów, gdyż nie wymaga ponownej kompilacji całego programu.

Trzeci przykład pokazuje, jak łatwo można umieszczać napisy w wybranym obszarze ekranu nie troszcząc się o aktualne położenie kursora (ani go nie zmieniając).

```
procedure komunikat ( kom : wiersz );
  var i : byte;
begin
for i := 1 to Length(kom) do linia[i].znak := kom[i]
end;
```

Ta procedura może być szczególnie wygodna do umieszczania krótkich komunikatów w 25 wierszu ekranu.

Fantazji czytelników pozostawię perspektywy otwierające się przed programistą dzięki możliwości dostępu do atrybutów każdego znaku na ekranie z osobna. Wszelkie rozjaśnienia istniejących napisów, zmiany tła, podkreślenia, migotania itp. stwarzają nieograniczone wręcz możliwości uzyskiwania efektów specjalnych podnoszących walory wizualne pisanych programów.

Na koniec wrócę do sprawy ekstra pamięci na kartach wyświetlaczy ekranowych. Hercules niestety nie daje zbyt wielkiego pola do popisu. Dodatkowe bajty można wykorzystać jako jeszcze jeden blok pamięci, ale zazwyczaj wcale nie jest to konieczne - jest napraszaniem się o kłopoty w przyszłości. Ciekawsza pod tym względem jest karta CGA, gdyż w BIOS-ie przewidziano funkcję, pozwalającą w trybie tekstowym przełączać strony pamięci ekranowej. 16Kb na karcie CGA podzielone jest na cztery strony (w trybie 80 kolumn lub na 8 stron w trybie 40 kolumn), z których każda może być aktywna. Przełączanie ekranów odbywa się błyskawicznie i nie ma potrzeby przechowywania ich aktualnej zawartości. Wpisywać informację można oczywiście na każdą stronę, niezależnie od tego czy jest aktualnie widoczna (o ile oczywiście zgodzimy się na bezpośrednie grzebanie w pamięci). Strony numerowane są od 0 do 3. Celem aktywizacji którejś z nich wystarczy umieścić jej numer w rejestrze AL i wygenerować przerwanie nr 16 ($10).

W starszych modelach karty CGA przy bezpośrednim zapisie w pamięci ekranu może pojawiać się nieprzyjemny efekt "śnieżenia". Jak go unikać, pisał Roland Wacławek w 12 numerze "Komputera" w roku 1987 w artykule "Turbo Pascal i karta CGA".

Zachęcam wszystkich do eksperymentów z pamięcią ekranu. Kryje się w niej nadspodziewanie wiele ciekawych możliwości.

\W pracy\

Mariusz Dec

Norton Integrator

Jeden z najznakomitszych programistów komputera IBM PC, Peter Norton wpadł na świetny (jak zwykle) pomysł. Wybrał swoje "sztuczki i chwyty", opatrzył je komentarzem i napisał program, który pozwala bardzo wygodnie z nich korzystać.

Program NORTON INTEGRATOR (NI.EXE) jest zbiorem instrukcji obsługi dwudziestu dwóch małych i większych programów, które mieliśmy już wcześniej. Mieliśmy, lecz nie znaliśmy - okazuje się bowiem, że większość tych programów można wykorzystać na tyle sposobów, że bez opisu ani rusz.

Zatem do dzieła!. **NI, Enter** i już po chwili ukazuje się charakterystyczna ramka. Po prawej stronie na dole możemy przeczytać, że nie musimy krzyczeć HELP! - wystarczy nacisnąć F1 i zobaczymy ekran pomocniczy zawierający informacje o obsłudze programu. Z informacji tam wymienionych chciałbym zwrócić uwagę na **Speed Search** - szybkie wyszukiwanie. Program korzysta ze standardowych funkcji systemu BIOS przy pisaniu tekstu na ekranie, a zatem kolejne ekrany ściągawek przewijają się denerwująco długo, o ile nasz BIOS nie jest zbyt sprawny. Pomogłaby tu wymiana pamięci ROM BIOS, czego ze zrozumiałych względów wszystkim nie mogę polecić lub użycie trybu **Speed Search**, co mogę polecić wszystkim.

Lewa strona ekranu wyświetla część listy dostępnych programów, prawa instrukcję obsługi wyróżnionego programu. Uważna lektura dostarcza kompletu informacji, lecz jak się okazuje, nie wszyscy czytają wystarczająco umiejętnie, aby dowiedzieć się wszystkiego. Za to, gdy już się dowiedzą, piszą o tym dla innych - po co mamy popełniać te same błędy...

Pierwszy na liście figuruje program **ASK.EXE** - bardziej przydatny niż się w pierwszej chwili wydaje. Umożliwia on tworzenie interakcyjnych plików wsadowych. Dzięki temu, po włączeniu komputera, można przenieść się do określonego programu czy wymaganej aplikacji naciskając tylko jeden klawisz z listy! Odpowiednio przygotowany plik wsadowy **(*.BAT)** uruchamiany z pliku **AUTOEXEC.BAT** jest bardzo wygodny, gdyż umożliwia łatwe zestawianie potrzebnych w danym momencie nakładek systemowych - zwalnia nas od nerwowego wciskania CTRL-C podczas uruchamiania systemu (przerwanie AUTOEXEC.BAT). Przydaje się to głównie wtedy, gdy z jednego komputera z twardym dyskiem korzysta wiele osób. Przygotowany w ten sposób plik wsadowy w jednym z redakcyjnych komputerów zakończył problemy ze zmieniającą się z każdym kolejnym użytkownikiem zawartością AUTOEXEC.BAT i... przetrwał w praktycznie niezmienionej formie ponad pół roku! Zadziwiające!

```
The Norton Integrator

You can execute any the Norton Utilities from within NI: Move the
highlight to the command, type any arguments (optional), and press
Enter

Tab             Erase command line and enter Speed Search mode
Enter           Run the program on the command line
Space           Leave Speed Search
Up, Down        Move to previous or next command
Letters         Type arguments on the command line
Esc, F10        Quit NI

Comments

Speed Search    Typing letters moves the highlight to the first
                program that matches the letters typed so far.
                Press Enter to run the command, or Space to add
                arguments

[argument]      In the help screens, arguments in brackets are optional

                Press any key to continue.
```

Drugi na liście program - o nazwie **BEEP.EXE** - daje możliwość wykorzystania naszego PC jako syntezera dźwięku. Podane np. w postaci pliku tekstowego dane umożliwiają komponowanie prostej muzyki.

Amatorzy porządków na pewno chętnie skorzystają z programu **DS.EXE**, który umożliwia sortowanie katalogu według wybranego klucza. Z kolei wybór katalogu można przeprowadzić (z poziomu programu NI) za pomocą programu **NCD.EXE**, który może być ponadto przydatny, gdy chcemy przyjrzeć się strukturze katalogów na naszym dysku twardym. Uznanie dla Petera Nortona wzmaga fakt, że zapis zmiany katalogu **CD \ KAT1 \ KAT2 \ ... \ KAT13** możemy zastąpić prostym: **NCD KAT13.**

W tym miejscu chciałbym zwrócić uwagę na zbiór danych o podkatalogach tworzonych przez ten program - **TREEINFO.NCD.** Gdy został utworzony nowy katalog, a nie została zapisana informacja o tym, program nie będzie mógł z niego korzystać.

Program **DT.EXE** testuje dysk pod kątem obecności na nim sektorów wadliwych fizycznie (np. uszkodzenie nośnika). Szczególnie przydatna jest możliwość odszukania i zaznaczenia bloków (ang. *cluster*), które uległy uszkodzeniu podczas eksploatacji dysku twardego. Należy bowiem pamiętać, że MS-DOS ma możliwość zaznaczenia wadliwych bloków jedynie podczas formatowania.

Program **FA.EXE** jest pomocny w chwili, gdy zależy nam na nadaniu specjalnych atrybutów zbiorom dyskowym. Nie różni się szczególnie od systemowego programu **ATTRIB.EXE**, lecz opis po prawej stronie ekranu NI pomaga, a co najważniejsze zwalnia od konieczności zaglądania do dokumentacji.

Twardy dysk zawiera zwykle tysiące zbiorów. Może się zdarzyć, że zapomnimy, gdzie znajduje się zbiór lub program, który jest nam potrzebny. Program **FileFind - FF.EXE** wyręczy nas z przeglądania katalogów "na piechotę". Zagubiony zbiór to nic w porównaniu ze zgubioną złotą myślą, którą kiedyś zawarliśmy w zbiorze dyskowym, a nie pamiętamy, w którym. Norton pomyślał i o tym. Program **TextSearch (TS.EXE)** pracowicie przejrzy wszystkie zbiory oraz, gdy zechcemy, także obszary dysku po skasowanych zbiorach. Ciekawostką jest tutaj możliwość wyboru początkowego bloku, od którego będzie przeszukiwany dysk.

```
The Norton Integrator

ASK                  ASK         ASK "prompt", [key-list]
BEEP                    Create interactive batch files.
DS  Directory Sort
DT  Disk Test           When ASK is run, it displays the prompt, then
FA  File Attributes     awaits a response. Respond by typing one of the
FF  File Find           keys in key-list, and ASK returns control to
FI  File Info           the batch file, passing the chosen key as an
FR  Format Recover      ERRORLEVEL code: the 1st key as level 1, the
FS  File Size           2nd as level 2, and so on. Arrange the branches
LD  List Directories    in descending order of ERRORLEVELs-check for
LP  Line Print          the highest level first, then next highest,
NCD Norton CD           etc., so that ERRORLEVEL 1 is checked for last.
NU  Norton Utility
QU  Quick UnErase    Example batch file
SA  Screen Attributes      Ask "Press "Y" to answer Yes", ny
SD  Speed Disk             If errorlevel 2 goto run
SI  System Information     If errorlevel 1 goto quit
TM  Time Mark              :run
TS  Text Search            echo You pressed "Y"
UD  UnRemove Directory     :quit
               more...
ASK
                                              Press F1 for Help
```

Gdy mowa o pamięci dobrej lecz krótkiej, warto zwrócić uwagę na program **FI.EXE.** Jest to uzupełnienie pomysłu znanego z Norton Commandera (opis katalogu zawarty w zbiorze tekstowym **DIRINFO**) - w pliku **FILEINFO.FI** możemy umieścić opisy zbiorów. Każdy katalog może zawierać osobny opis dotyczący zbiorów w nim zawartych. Wywołanie programu bez parametrów jest równoważne z poleceniem DIR, a do wyboru mamy 8 różnych opcji programu.

Czy można zabezpieczyć się przed przypadkowym formatowaniem dysku twardego? Częściowe zabezpieczenie umożliwia program o nazwie **Format Recover (FR.EXE).** Przepisuje on FAT (tablicę alokacji) oraz katalog w "końcowy" obszar dysku - najbliżej fizycznego środka. Dzięki temu omyłkowo sformatowane pierwsze ścieżki dysku, które zawierają FAT i katalog główny, mogą zostać odtworzone, a zbiory fizycznie nie skasowane - odzyskane.

Informacje o zajętej lub wolnej przestrzeni dysku czasami się przydają. Dokładna specyfikacja uwalnia nas od rachunków, a przy okazji wyświetlane są katalogi całej wybranej ścieżki - program **FS.EXE (FileSize).** Natomiast drzewo katalogów całego dysku można uzyskać za pomocą programu **LD.EXE** (ang. *List Directories*).

> 38

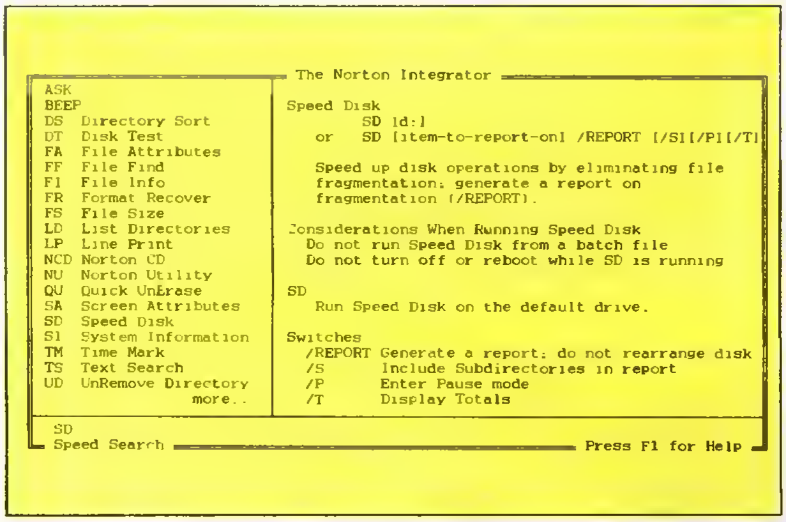
37 <

Wart uwagi jest również program **LP.EXE** - można go polecić do przygotowania wydruków programów uzupełnionych o numery linii, nagłówki itp. Opcji jest znów kilkanaście, a interesującą cechą - możliwość wydruku na drukarce pracującej w standardzie ASCII plików zapisanych w standardzie EBCDIC (IBM 360).

Cały omawiany pakiet potocznie nazywany jest *Norton Utilites* (i "Komputer" nie ustrzegł się tego błędu), gdy tymczasem **program NU.EXE jest solidnym, samodzielnym systemem dla "dłubacza dyskowego"**.

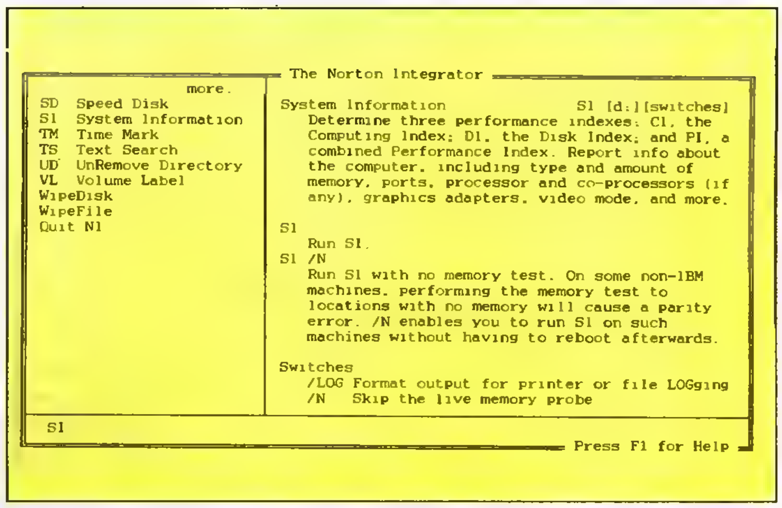
Program niewątpliwie zasługuje na oddzielny opis, lecz jak zwykle w przypadku programu umożliwiającego poważne zmiany na dysku, mam wątpliwości czy rzeczywiście jest to narzędzie dla każdego użytkownika komputera PC. Posługując się nim można np. usunąć lub zmienić "partycję" dysku twardego, skopiować bloki, wprowadzić zmiany w tablicy *FAT* i katalogu. Działalność taka wymaga rzetelnej wiedzy o dysku i jego organizacji - zdarzyło mi się ratować dysk po ingerencji za pomocą **NU.EXE**. Uważam ten program za prawie idealne narzędzie do prac na dysku, myślę jednak, że część funkcji tego programu, dostępna w zestawie NI, jest zdecydowanie bezpieczniejsza w użyciu.

Myślę, że wielokrotnie zdarzyło się nam popełnić błąd i niepotrzebnie usunąć zbiór z dysku. Odzyskanie takiego zbioru jest możliwe, dopóki nic nie zostało zapisane w miejscu przez niego zwolnionym. Możliwe nie oznacza, że łatwe. Czasami trzeba przejrzeć wiele bloków zanim natrafi się na brakujący fragment. W omawianym zestawie rolę poszukiwacza spełnia program **QU.EXE** (ang. *Quick UnErase*). Sukces w odszukiwaniu skasowanych zbiorów zależy od porządku na naszym dysku. Co należy rozumieć przez porządek? Otóż, jak wiadomo, każdy zbiór opisany jest w tablicy alokacji *(FAT)* będącej łańcuchem wskaźników bloków *(cluster)* zawierających następne fragmenty. Gdy nasze zbiory umieszczone są w kolejnych blokach dysku, nie ma problemu z ich odzyskaniem. Gdy bloki są przeplecione i rozrzucone po dysku - ich automatyczne odszukanie nie jest możliwe i pozostaje żmudne przeglądanie dysku. Co więc należy zrobić, aby kolejne bloki zbioru były kolejnymi blokami dysku? Należy używać programu **SD.EXE - Speed Disk** - nazwa ta wskazuje na "efekt uboczny" porządkowania zbiorów według podanej wyżej zasady. Istotnie, zbiory zapisane w kolejnych blokach odczytywane są zdecydowanie szybciej - głowica nie musi się daleko przemieszczać podczas odczytu.



Program **SD.EXE** przepisuje zawartość dysku, układając zbiory w kolejne bloki. Najbardziej zauważalne jest zdecydowane wyciszenie napędu głowic oraz zwiększenie szybkości pracy dysku twardego. Rzecz jasna, efekt jest tym lepiej widoczny, im wolniejszym dyskiem dysponujemy. Na temat użytkowania tego programu krąży szereg niekonkretnych opinii, do których (niestety) nasza redakcja też się przyczyniła. Podstawowy problem dotyczy obecności nakładek w pamięci. Nie powodują one błędnej pracy programu samą swoją obecnością, ale prawie na pewno "narozrabiamy", gdy zechcemy za pomocą **SideKicka** edytować zbiór i zapisać go na dysk podczas pracy SD. To samo dotyczy **Super Key** i dowolnego innego programu przerywającego proces podstawowy i przejmującego sterowanie systemem dyskowym. Pamiętajmy, że omawiany program przepisuje praktycznie całą zawartość dysku - niespodziewany zapis w przygotowane wolne miejsce nie może się skończyć dobrze.

W przypadku dysku twardego 20 MB i komputera XT cała operacja może trwać nawet do 30 minut. Należy zachować cierpliwość i liczyć na to, że elektrownia nie wyłączy prądu. Nakładki znajdujące się w pamięci zwalniają proces przede wszystkim poprzez zmniejszenie bufora w pamięci i ogólne spowolnienie systemu. Czołowym "zwalniaczem" jest tutaj **SideKick** - efekt jego obecności widoczny jest w **dBase III** już w przypadku obróbki bazy o kilkuset rekordach indeksowanych. Gdy weźmiemy pod uwagę fakt, że **SideKick** obsługuje "po swojemu" aż 10 przerwań systemu BIOS - spowolnienie przestaje dziwić.



W związku z faktem przepisania zawartości dysku w nowe miejsce zmianie ulega fizyczna lokalizacja zbiorów. Słyszałem natomiast o "pomysłach" zabezpieczania programów instalowanych na twardym dysku poprzez zapis fizycznej lokalizacji do zbioru. Jeżeli nie zabezpieczy się takiej "kontroli" przed przepisaniem w nowe miejsce (zabezpieczenie takie jest możliwe) - program nie będzie funkcjonował. Przestrzegam zatem użytkowników programów z takimi "zabezpieczeniami" przed ich stosowaniem.

Gdy poważne zajęcia mamy już za sobą, możemy sprawdzić czy łatwo czyta się czerwone litery na niebieskim tle. Atrybuty i kolory ekranu pomoże nam ustawić program **SA.EXE - Screen Attributes**. Gdy kolory są odpowiednio ustawione, to nawet niedyskrecja programu **SI.EXE - System Information** - nie spowoduje ich ujawnienia. Natomiast gdy napisy będą w kolorze innym niż tło, dowiemy się, czy BIOS komputera jest licencjonowany, czy nie, jakie są adresy rozszerzeń ROM BIOS- u i kilku jeszcze szczegółów z życia wewnętrznego naszego PC.

Znacznik czasu **TM.EXE - Time Mark** - umożliwia odczyt czasu i jego pomiar pomiędzy kolejnymi wywołaniami programu. Dostarcza także wydruki - coś w rodzaju karty zegarowej, a do dyspozycji mamy 4 niezależne liczniki.

Nasze dobre samopoczucie psuje program **UD.EXE - UnRemove Directory** - po raz kolejny udowadnia, że nie zawsze wiemy, co robimy - skasowaliśmy katalog, a teraz chcemy go odtworzyć. Operacja ta jest niezbędna także w przypadku odtwarzania zbioru z usuniętego katalogu.

Gdy znudzi się nam etykieta (ang. *label*) dysku, na którym pracujemy, możemy ją zmienić za pomocą programu **VL.EXE - Volume Label**. Podobnie jak niektóre wymienione wyżej, również ten program ma swój odpowiednik w zestawie programów systemowych, ale kilka szczegółów, zwanych potocznie "bajerami", także może się przydać.

Ostatnie już na liście programy to **WIPEDISK.EXE** oraz **WIPEFILE.EXE**. Za ich pomocą możemy zamazać na dysku zbiory, których zawartość jest poufna. Zamazywanie oznacza wypełnianie całych sektorów, w których zapisany był zbiór, bajtem o jednakowej wartości (np. 0). Szczegóły zawarte są w opisie, warto jednak zwrócić uwagę na to, że program **WipeDisk** z podanym tylko określeniem napędu powoduje wyczyszczenie CAŁEGO dysku. Wytarcie obszaru nieużywanego wymaga podania opcji /E.

Na tym kończy się lista programów zestawu NORTON INTEGRATOR. Większość podanych wyżej uwag odnosiła się do dysku twardego. Tym, którzy mają taki dysk, zalecam otwarcie ścieżki dostępu do podkatalogu zawierającego NI i wykorzystywanie elementów zestawu w codziennej pracy.

Wiele z wymienionych funkcji realizują także inne programy (np. **PcTools**), lecz elementy omawianego pakietu można wykorzystywać osobno, co w wielu przypadkach bardzo przyspiesza pracę.

\W pracy\

Zenon Rudak

Kolgar portable AT

Kolejnym testowanym obiektem jest komputer standardu IBM PC, Kolgar AT w wersji przenośnej. Komputer otrzymaliśmy do testowania od pana Macieja Kollo, właściciela wysyłkowej firmy Kolgar, Bomenrijk 31, 1112 EL DIEMEN, Holandia, tel: 0-031-20952033, telefon warszawski 474581. Dziękujemy!

Komputer *Kolgar AT* to trzeci komputer typu IBM PC/AT testowany w tej rubryce. Tym razem jest to maszyna przenośna. Na ilustracji obok pokazano komputer złożony i przygotowany do transportu. Obudowa wykonana jest w postaci średniej wielkości walizki, której podstawę stanowi płyta z klawiaturą. Płytę zamocowano do obudowy za pomocą dwóch przesuwnych zatrzasków wykonanych z tworzyw sztucznych (podobnie cała obudowa składająca się z czterech przesuwanych i zdejmowanych pokryw). Pokrywa umieszczona w lewej bocznej ścianie zabezpiecza w czasie transportu gniazda przyłączenia sieci zasilającej i wlot powietrza do kanału wentylatora. Pokrywa prawa zasłania gniazda interfejsów, złącze monitora i inne gniazda lub przełączniki umieszczone na blaszanych wspornikach kart rozszerzenia. Pokrywa górna jest zdejmowana i umożliwia dostęp do gniazd połączeniowych kart rozszerzenia, znajdujących się na płycie głównej komputera. Po zdjęciu tej pokrywy użytkownik uzyskuje dostęp do potencjometrów regulacyjnych monitora i przełączników umieszczonych na płycie głównej. Druga taka pokrywa - na spodniej stronie obudowy - pozwala na wgląd do płyty głównej komputera od strony końcówek lutowniczych.

Plastykowa obudowa stanowi poszycie przestrzennie ukształtowanego blaszanego chassis, do którego mocowane są wszystkie elementy składowe maszyny. Chassis to element nośny i wiążący cały komputer.

Kilka słów o budowie komputera *Kolgar portable AT*. Maszyna zaprojektowana jest tak, aby mieściła w jednej, maksymalnie zmniejszonej obudowie wszystkie niezbędne części typowego komputera standardu IBM PC/AT. Obudowa jest nieznacznie większa od typowej "dużej" skrzynki komputerów tego standardu. Wewnątrz mieszczą się: płyta główna typu "baby 286", zasilacz sieciowy, napęd dysku twardego z krążkami o średnicy 5,25 cala, jeden napęd dyskietek elastycznych 5,25 cala, monitor monochromatyczny o przekątnej ekranu 23 cm (9 cali) i pełnowymiarowa klawiatura typu AT. Płyta główna wyposażona jest w procesor Intel 80286, 1024 KB pamięci RAM oraz w siedem złączy do mocowania kart rozszerzenia. Cztery z nich są 16-bitowe i przeznaczone dla kart AT lub XT, pozostałe - 8-bitowe, umożliwiające instalowanie kart typu XT. Jedno złącze AT zajęte jest przez płytę sterownika dysku twardego i napędów dyskietek. Zajęte jest także jedno złącze XT - przez

> 40



komputer test

39 <

kartę graficzną typu Hercules. Praktycznie użytkownik może wykorzystać trzy złącza typu AT i dwa typu XT. Elementy elektroniczne, z których zbudowana jest płyta główna i karty instalowane w komputerze, pochodzą z renomowanych firm japońskich, jak NEC, Mitsubishi, Chips. Zastosowano układy pamięci RAM firmy Sharp, napęd dyskietek elastycznych firmy NEC. (Jest to stacja 80-ścieżkowa, umożliwiająca zapisanie do 1,2 MB informacji.) Napęd dysku twardego umieszczony w komputerze *Kolgar* to wyrób firmy Seagate typ ST 225 o czasie dostępu ok. 40 ms. Napęd umożliwia przechowanie do 21 MB informacji.

Test

Instalacja i uruchomienie komputera jest proste i polega na odłączeniu płyty klawiatury od obudowy i podłączeniu przewodu zasilającego. Czynności te wykonuje się bardzo szybko i prosto, co ułatwia zmianę miejsca pracy komputera.

Dostęp do wnętrza maszyny jest łatwy i umożliwia swobodne instalowanie dodatkowych kart rozszerzenia. Dostęp do gniazd interfejsów również nie sprawia żadnych kłopotów, wystarczy przesunąć pokrywę w prawej bocznej ściance obudowy, aby odsłonić wszystkie złącza. Składanie maszyny jest łatwe i nieskomplikowane. Do przenoszenia komputera służy solidny uchwyt typu walizkowego zamocowany do blaszanego szkieletu wewnętrznego całej maszyny.

Przy testowaniu komputerów klasy IBM PC zwykle zwraca się uwagę na kilka spraw. Są to: jakość i sposób pracy klawiatury, rodzaj i jakość monitora, zgodność sprzętowa i programowa z oryginałem. Zacznę od początku.

Klawiatura

Komputer, mimo iż jest urządzeniem przenośnym i małym, ma pełną, normalnych wymiarów klawiaturę typu AT. Klawisze wymagają niewielkiej siły nacisku i pracują pewnie. Nie zauważyłem przekłamań czy zbyt szybkiego powtarzania swojej funkcji. Klawiatura umożliwia szybkie pisanie, oznaczenie klawiszy jest czytelne. Do zmiany kąta położenia klawiatury służą odchylane, umieszczone w spodniej części płyty klawiatury języczki. Zastosowanie gumowych nóżek zapewnia stabilność położenia klawiatury na biurku - klawiatura nie przesuwa się, gdy opiera się o nią ręce. Z komputerem połączona jest przewodem zwiniętym w sprężynę. Przewód ten chowa się do otworu obok napędu dyskowego, gdy komputer jest składany. Stan taki powoduje, że przewód nie jest zakończony wtyczką łączącą - klawiatura połączona jest z komputerem na stałe.

Monitor

W komputerze *Kolgar portable AT* zastosowano monitor monochromatyczny o bursztynowej barwie obrazu. Ekran jest stosunkowo mały. Monitor sterowany jest z karty graficznej Hercules. Małe wymiary ekranu i duża liczba wyświetlanych punktów (wysoka rozdzielczość karty Hercules) zapewniają bardzo dobrą czytelność i wyrazistość obrazu graficznego i tekstowego. Zastosowany monitor może być użyty także jako ekran dla kolorowej karty graficznej. W testowanym egzemplarzu komputera monitor miał przesunięty obraz w lewo o ok. 2 mm. Usterki tej nie można było usunąć potencjometrami dostępnymi po zdjęciu pokrywy górnej komputera. Wada ta praktycznie nie ma znaczenia, gdy wykorzystywany jest tryb tekstowy sterownika obrazu. Przejście w tryb graficzny powoduje gubienie lewej krawędzi obrazu. Sytuację taką można poprawić odpowiednio regulując układy elektroniczne monitora. Regulacja ta wymaga jednak wymontowania monitora z obudowy komputera, co jest dość trudne ze względu na zwartość budowy maszyny. Pozostałe parametry monitora były bez zarzutu. Obraz graficzny bez zniekształceń i zakłóceń, zachowana równoległość i prostopadłość linii, nie występowało spłaszczenie obrazu ani widoczne zmiany jego gęstości w różnych miejscach ekranu.

Monitor sterowany jest, jak wspomniałem wcześniej, z karty graficznej typu Hercules - karty typu "krótkiego". Umieszczona w testowanym egzemplarzu komputera karta miała zbyt krótkie złącze. Nie zapewniało ono pewnego połączenia w gnieździe płyty głównej. Zdarzały się przypadki utraty połączeń i w efekcie brak obrazu lub blokada całej maszyny. Sytuacje takie występowały najczęściej wówczas, gdy komputer był przewożony. Wyjętą z testowanego komputera kartę graficzną montowałem w kilku innych komputerach. Karta pracowała poprawnie wszędzie tam, gdzie złącza płyty głównej były ciasne. Montowałem także inne karty graficzne w komputerze *Kolgar AT*. Dawały się one bez problemu montować i współpracowały poprawnie z monitorem komputera.

W wyniku moich sugestii i dyskusji z osobami prowadzącymi serwis, właściciel firmy postanowił sprowadzać do Polski karty wizji w wykonaniu "długim", umożliwiające lepsze i pewniejsze mocowanie karty w gnieździe i komputerze.

Zgodność sprzętowa

Dla sprawdzenia zgodności sprzętowej zamontowałem w komputerze kartę obsługi *Handy Scannera* (testowany w nr. 4/88). Program obsługi uruchomił skaner bez najmniejszych kłopotów. Możliwe było wykorzystanie wszystkich opcji sprzętowych i programowych. Drugim elementem dołączonym do komputera była karta modemu telefonicznego. Karta ta pracowała bez zarzutu i pozwalała na poprawne przeprowadzanie łączności między komputerami za pomocą łączy telefonicznych. Sprawdziłem także działanie interfejsu równoległego typu Centronics służącego do podłączenia drukarki. Interfejs sterował drukarkami dając bezbłędne wydruki tekstów i grafiki. Nie sprawdziłem możliwości transmisji zbiorów między testowanym komputerem a innymi maszynami z powodu braku interfejsu szeregowego w wyposażeniu komputera.

Dostarczony do redakcji komputer wyposażony był w pamięć RAM o pojemności 1024 KB. Obszar ten podzielono na dwie jednakowe części po 512 KB. Część pierwszą przeznaczono dla systemu operacyjnego DOS i programów, drugą na RAM-dysk. W czasie testowania usiłowałem zmienić stosunek podziału pamięci RAM tak, aby uzyskać 640 KB dla systemu i programów, a resztę przeznaczyć na RAM-dysk. Przełączanie dostępnych na płycie głównej i opisanych w instrukcji obsługi przełączników nie dało rezultatu. Zmiana parametrów ustalanych w obszarze pamięci RAM podtrzymywanym z baterii (kilkadziesiąt bajtów pamięci RAM umieszczonych w układzie zegara czasu astronomicznego) za pomocą programu służącego do konfiguracji komputerów typu AT (IBM SET UP) też nie dała rezultatu. Przełączanie przełączników podziału pamięci umożliwia uzyskanie obszaru o pojemności 640 KB dla systemu operacyjnego, ale nie jest "widoczny" wtedy RAM-dysk.

Użyta do budowy komputera płyta główna jest konstrukcją wykorzystującą dla pamięci operacyjnej obszar 512 KB RAM i jest zgodna z podstawowymi założeniami konstrukcyjnymi komputera AT. W zasadzie taki stan nie przeszkadza w poprawnej pracy komputera, ale mogą zdażać się sytuacje, gdy pojemność pamięci operacyjnej będzie zbyt mała. Szczególnie wówczas, gdy używamy rezydujących nakładek systemowych.

Po przeprowadzonych testach wydaje się, że zgodność sprzętowa jest zadowalająca i nie będzie kłopotów z wykorzystaniem dodatkowych urządzeń rozbudowujących komputer.

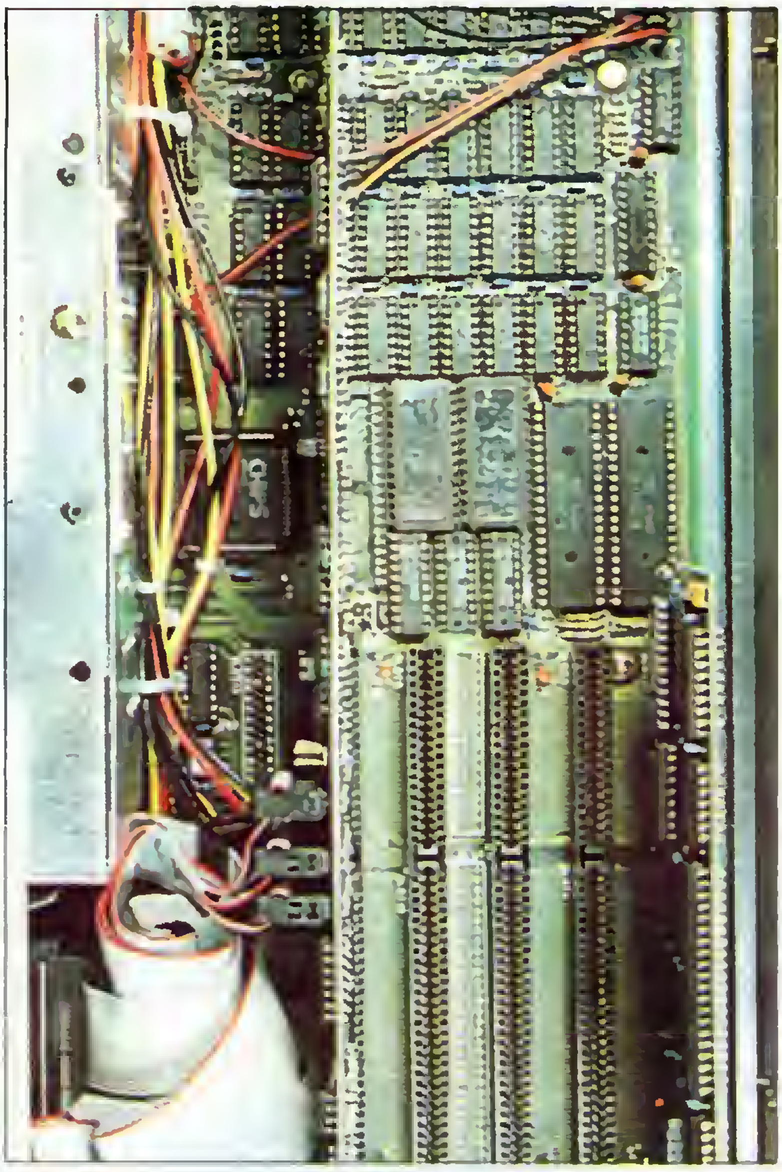
Po pewnym okresie pracy maszyny pojawiło się niepokojące zjawisko. Otóż po włączeniu zasilania test pamięci RAM, wykonywany przez procedury startowe znajdujące się w pamięci ROM z systemem BIOS komputera, wykazywał wystąpienie błędu parzystości pamięci operacyjnej. Błąd występował po sprawdzeniu pierwszych 64 KB pamięci (bank 0). Ponieważ obszar ten jest niewystarczający dla uruchomienia systemu operacyjnego, praca maszyny była niemożliwa. Po kikakrotnym naciśnięciu przycisku Reset lub kilku próbach wyłączenia i włączenia komputera wada ustępowała. Zjawisko to występowało losowo. Wadę tę powodowało odkształcenie płyty głównej komputera, spowodowane zbyt silnym dokręceniem śrub mocujących płytę do chassis. Nagrzewanie się układów lutowanych do płyty dodatkowo pogarszało sytuację. Występująca wada była wadą typowo mechaniczną niezależną od dostawcy. Sprawnie pracujący serwis firmy Kolgar szybko usunął wszystkie występujące i opisane przeze mnie usterki. Niedomagania usunięte przez serwis nasuwają pewne spostrzeżenie. Wydaje się, że w dobrze pojętym interesie firm i ich klientów trzeba by pomyśleć o wprowadzeniu przeglądu "zerowego" przed rozpoczęciem eksploatacji sprowadzanych urządzeń.

Obecnie firma Kolgar zorganizowała pełny serwis techniczny i kontroluje wszystkie komputery przed dostarczeniem ich do klientów.

Zgodność programowa

Testując komputer uruchamiałem szereg popularnych programów graficznych, baz danych, edytorów tekstu, kompilatory języków programowania (Turbo Basic i Turbo Pascal firmy Borland), programy narzędziowe i testujące. Wszystkie dawały się zainstalować i pracowały poprawnie. Kłopoty wystąpiły tylko z jednym krótkim programem o nazwie WARBLE.EXE. Program ten emituje z głośnika komputera dźwięk imitujący dzwonek telefonu. Program był odczytywany przez komputer i realizowany, ale bez efektu dźwiękowego. Działał on na wszystkich testowanych dotychczas

test KOMPUTER



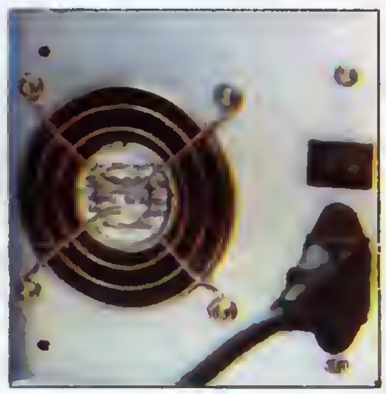
przeze mnie komputerach standardu IBM PC. Mimo że dźwięk nie był emitowany, nie następowało zawieszanie się komputera i nie występowały jakiekolwiek oznaki zakłóceń pracy maszyny. W innych programach działanie głośnika było prawidłowe. W sumie po przetestowaniu można stwierdzić, że zgodność programowa jest dobra i użytkownicy nie będą mieli kłopotów z uruchamianiem programów i pracą komputera. Dodatkową gwarancją dobrej zgodności programowej z oryginałem jest komunikat ukazujący się na ekranie monitora w czasie uruchamiania komputera, że maszyna wyposażona jest w licencjonowany ROM z BIOS firmy Award Software Inc.

Procesor komputera *Kolgar portable AT* taktowany jest z zegara o częstotliwości 10 MHz. Maszyna jest bardzo szybka. Program testujący szybkość pracy maszyn standardu IBM PC SPEED.COM wykazał 6,2 raza większą szybkość pracy *Kolgara AT* niż oryginału IBM PC/XT. Tak duża szybkość pracy nie powoduje zakłóceń przy wyświetlaniu obrazu lub współpracy z interfejsami. Zastosowany dysk twardy firmy Seagate ma czas dostępu ok. 40 ms. W połączeniu z szybko działającym komputerem zapis, odczyt czy wyszukanie wybranych zbiorów są prawie natychmiastowe. Duża szybkość przetwarzania informacji znacznie skraca działanie wielu programów np: sortujących, nakładkowych, przetwarzających obrazy graficzne itp.

Tak jak każdy komputer standardu PC *Kolgar AT* wyposażony jest w wentylator wymuszający chłodzenie elementów elektronicznych i zasilacza sieciowego. Poziom emitowanego hałasu jest zbliżony do tego, jaki wytwarzają podobne mu kopie IBM.

W testowanym komputerze zainstalowany był tylko jeden napęd dyskietek elastycznych. Stacja umożliwiała zapisywanie dyskietek do pojemności 1,2 MB (80 ścieżek po 15 sektorów 512-bajtowych) i transmisję danych z szybkością 62,5 KB na sekundę. Wielokrotnie wspominałem o pewnych kłopotach przy pracy z taką stacją z dyskietkami używanymi także w komputerach XT, gdzie zapis jest 40-ścieżkowy. Uważam za konieczne instalowanie w komputerach PC/AT dwóch stacji dyskietek, jednej 80-, drugiej 40-ścieżkowej. Taki układ pozwala na bezawaryjną pracę i swobodne przenoszenie dyskietek ze zbiorami między komputerami AT i XT. Drugim ważnym elementem jest wyposażenie komputera w interfejs szeregowy typu RS 232 C. Brak takiego interfejsu uniemożliwia najprostszy wariant połączenia ze sobą komputerów oraz korzystanie z manipulatorów typu mysz, digitizerów, ploterów - urządzeń niezbędnych przy pracy z większością programów graficznych.

Komputer *Kolgar AT* dostarczany jest wraz z opisem płyty głównej, karty sterownika ekranowego, karty sterownika napędów dyskowych, obsługi dysku twardego oraz książką zawierającą podstawowe informacje o komputerze PC/AT, opisy komend dyskowego systemu operacyjnego oraz zasady programowania w języku Basic. Dostarczona literatura pozwala użytkownikowi na sprawdzenie poprawności ustawienia podstawowych przełączników konfiguracyjnych płyty głównej oraz na poprawne zainstalowanie dodatkowych kart rozszerzenia. Opisy komend systemu operacyjnego podają tylko niezbędne informacje dotyczące sposobu ich zapisu i działania.





Na marginesie tego testu pociesza fakt podejmowania dyskusji i wyciągania wniosków przez kierownictwo firmy z informacji dostarczanych przez klientów (także moich wynikających z testów) a mających znaczenie dla jakości oferowanego sprzętu i sprawności firmy. Mam nadzieję, że nastąpi ogólna moda na taki styl działania.

Zalety komputera *Kolgar portable AT:*
- bardzo szybka praca;
- szybki dysk twardy;
- wyraźny monitor;
- dobra klawiatura;
- zwarta budowa (cała maszyna w jednym pudełku).

Wady komputera *Kolgar portable AT:*
- brak interfejsu RS 232 C;
- brak 40-ścieżkowej stacji dyskietek (od PC/XT);
- nieodłączalna klawiatura.

Dane techniczne komputera *Kolgarportable AT:*

procesor	Intel 80286;
zegar procesora	10 MHz;
pamięć RAM	1024 KB, z tego 512 KB dla pamięci operacyjnej i 512 KB dla RAM-dysku;
pamięć ROM	32 KB z BIOS komputera;
pamięć zewnętrzna	1 napęd dyskietek elastycznych 5,25 cala zapisywanych do pojemności 1,2 MB (80 ścieżek po 15 sektorów 512-bajtowych); 1 napęd dysku twardego o pojemności 21 MB;
interfejs	równoległy typu Centronics;
karta graficzna	Hercules (mono o wysokiej rozdzielczości);
monitor	o bursztynowej barwie ekranu i przekątnej 23 cm (9 cali);
klawiatura	pojemnościowa typu AT o 84 klawiszach;
wymiary	długość 49 cm; wysokość 19 cm; szerokość 47 cm;
waga	ok. 23 kg.

KOMPUTER test

\W pracy\

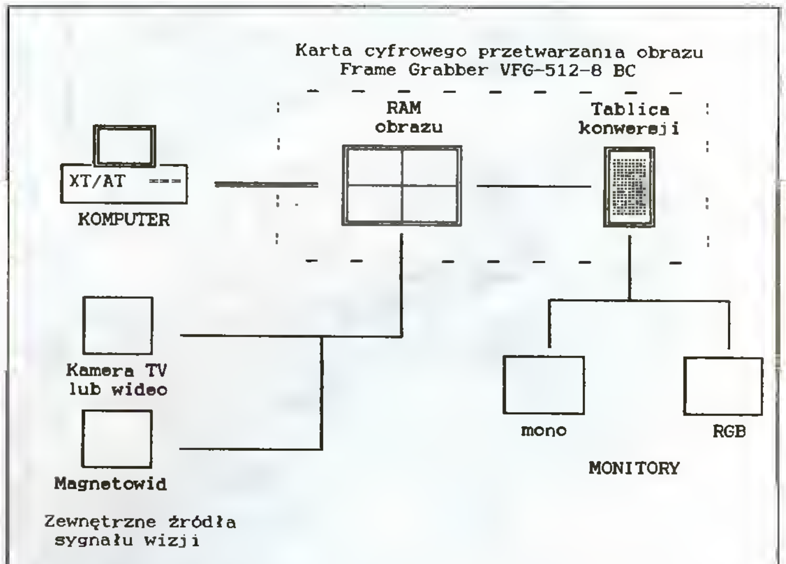
Zenon Rudak

Frame Grabber

W wielu profesjonalnych zastosowaniach komputerów obraz jest podstawą analiz, wnioskowania, jest często jedynym dokumentem zaistniałych zdarzeń. Najczęściej obraz taki powstaje w urządzeniach kontrolno-pomiarowych lub w czasie wykonywania eksperymentów na obiektach badań. Nowoczesna technika pozwala rejestrować takie obrazy za pomocą techniki telewizyjnej. Zapis obrazu nie zawsze jednak daje się łatwo dalej obrabiać lub archiwizować. Wybranie interesujących jednostkowych obrazów zmusza do pracy w dużych i skomplikowanych studiach telewizyjnych. Aby można było archiwizować lub modyfikować takie obrazy bezpośrednio na stanowisku pracy, firma Visionetics opracowała kartę cyfrowej obróbki obrazu *Frame Grabber VFG-512-8BC* instalowaną w komputerach standardu IBM PC/XT/AT.

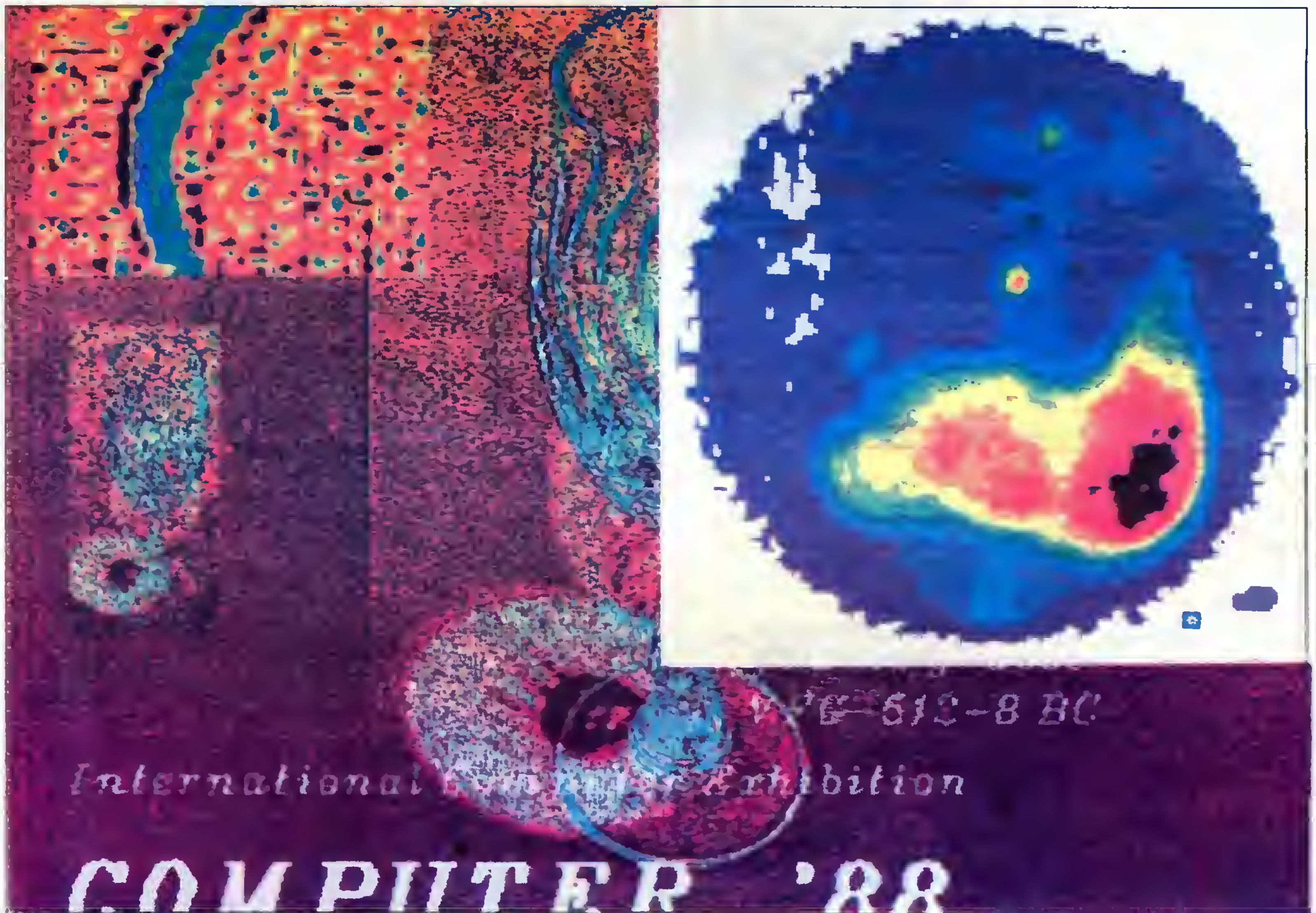
Zadania karty są następujące:

- przetwarzanie całkowitego sygnału wizyjnego na postać cyfrową;
- zapamiętywanie wybranych obrazów;
- przenoszenie obrazu do i z komputera;
- generowanie sygnału wizji dla monitora lub magnetowidu.



Rys. Schemat przebiegu sygnałów przy zastosowaniu karty cyfrowego przetwarzania obrazu typu Frame Grabber.

Karta przetwarza obraz pochodzący z kamery TV lub magnetowidu. Obraz pobierany z wejścia zamieniany jest na postać cyfrową, następnie zapisywany w pamięci RAM karty i przekazywany do układu konwersji i formowania wizyjnego sygnału wyjścia. Do wyjścia karty podłączony jest telewizyjny monitor analogowy niezależny od monitora komputera. Oprócz monitora do wyjścia karty można podłączyć urządzenia do zapisu obrazu. Karta pracuje niezależnie od komputera, nie zajmuje pamięci wizji komputera. Na żądanie operatora wybrany obraz może być przepisany do pamięci operacyjnej i poddany obróbce niezależnie od tego co dzieje się na wejściu karty.



Przykład wykorzystania karty Frame Grabber do kompozycji obrazu z kilku niezależnych fragmentów

Karta zapewnia rozdzielczość graficzną obrazu 512 na 512 punktów. W trybie monochromatycznym dysponuje 256 - stopniową skalą szarości. Pamięć RAM w tym trybie zajmuje 256 KB i podzielona jest na cztery bloki po 64 KB. W trybie kolorowym każdy punkt ekranu określony jest trzema kolorami podstawowymi: niebieskim, zielonym i czerwonym. Intensywność każdego z kolorów określa 64 - stopniowa skala. Pamięć obrazu kolorowego zajmuje 512 KB w ośmiu blokach po 64 KB.

Komputer współpracujący z kartą *Frame Grabber* ma bezpośredni dostęp do określonego bloku pamięci obrazu karty. Adres aktywnego bloku pamięci określa użytkownik przełączając odpowiednie przełączniki konfiguracyjne karty. Możliwość takiej konfiguracji pozwala na pracę jednego komputera i kilku kart tego typu jednocześnie i niezależnie od siebie. Komputer steruje kartą przez ustalanie zawartości rejestrów jej procesora obrazu. Obraz może być przetwarzany w sposób ciągły, zwolniony lub mogą być wybierane stop klatki.

Zwolnienie obrazu dokonywane jest przez opuszczanie założonej liczby zapisów pamięci obrazu. Zatrzymanie zapisu pozwala na ingerencję operatora w obraz. Poza takimi fotograficznymi metodami *Frame Grabber* umożliwia inne metody obróbki obrazu. Polegają one na częściowej aktualizacji wybranych punktów lub aktualizacji ich walorów. Umożliwia to zmiany kolorów elementów ruchomych obrazu, wykorzystanie skali szarości do określania intensywności obserwowanych zjawisk, nakładanie na obraz rzeczywisty elementów grafiki lub łączenie kilku obrazów ze sobą.

Obraz wysyłany z karty poddawany jest konwersji. Polega ona na przetworzeniu opisu wszystkich punktów obrazu za pomocą tablic konwersji ustalanych przez użytkownika. Tablice pozwalają na zamianę kolorów, zamianę skali szarości na skalę kolorów, porównywanie obrazu aktualnego z przechowywanym w pamięci komputera, zmianę pozytywu w negatyw, eliminację wybranych punktów lub barw z obrazu itp.

Karta *Frame Grabber* dostarczana jest z oprogramowaniem wykonującym następujące funkcje:

- ustalanie trybu pracy karty,
- definiowanie tablic konwersji,
- określanie sposobu pracy z obrazem rzeczywistym (spowalnianie, zatrzymanie, nakładanie),
- zapis i odczyt wybranego obrazu na nośniku pamięci zewnętrznej komputera,
- wydruk obrazu na drukarce (przewidziano użycie drukarki kolorowej),
- przenoszenie wybranych fragmentów między obrazami,
- przekształcenia geometryczne obrazów (zmiana wymiarów, obroty, kopiowanie),
- porównywanie obrazów zapisanych w pamięci komputera z obrazem rzeczywistym lub wzorcowym,
- filtracja tonalna obrazu,
- konturowanie elementów obrazu,
- wyznaczanie środków ciężkości, obliczanie powierzchni, określanie parametrów statystycznych obiektów obrazów,
- pomiary odległości między wyznaczonymi punktami obrazu.

Frame Grabber może mieć zastosowanie w wielu dziedzinach nauki i techniki. Szczególnie przydatna będzie w badaniach ośrodków naukowych, działalności laboratoriów analitycznych a także w wielu działach medycyny. Stanowisko komputerowe wyposażone w kartę cyfrowego przetwarzania obrazu tego typu może służyć do analizy zdjęć z mikroskopów elektronowych, zdjęć rentgenowskich narządów wewnętrznych człowieka, umożliwia analizę struktury powierzchni materiałów itp. Przy zastosowaniu kamer TV o liniowej charakterystyce odwzorowania skali szarości i wymiarów możliwa jest realizacja bardzo dokładnych systemów pomiarowych. Porównywanie obrazów rzeczywistych i wzorcowych, badanie rozkładu temperatur, graficzna analiza zdjęć rentgenowskich stwarzają warunki do diagnozowania i wykrywania organów objętych zmianami chorobowymi itp. Karty cyfrowego przetwarzania obrazów stosowane są w stanowiskach komputerowych obsługujących układy termowizyjne, sondy izotopowe i ultrasonograficzne.

Inną dziedziną zastosowań jest grafika komputerowa, tworzenie filmów animowanych, wspomaganie studiów telewizyjnych i filmowych.

Redakcja nasza miała możliwość zapoznania się z kartą typu *Frame Grabber* na pokazie zorganizowanym przez Przedsiębiorstwo Doradztwa i Wdrożeń Przemysłowych SIRPOL z Sieradza.

\Kurier\

Prosto z dysku

Dodatki do Turbo Pascala 4.0

Nie minęły trzy miesiące od pojawienia się nowej wersji Turbo Pascala, a już oferowane są programy rozszerzające jej możliwości, np. firma Turbo-Power Software proponuje Overlay Manager, TDebug 4.0 oraz Turbo PROFESSIONAL 4.0 - bibliotekę ponad 300 procedur rozszerzających funkcje TP4.

Firma Blaise Computing proponuje natomiast Power Plus Tools 4.0 - bibliotekę ponad 140 funkcji wzbogacających środowisko programowe użytkownika TP4, w tym rozszerzone możliwości pułapkowania i budowy programów rezydujących oraz rozbudowaną obsługę ekranu i pojawiających się na nim okien.

Inny program tej samej firmy - Asynch Plus - pomaga użytkownikowi TP4 tworzyć programy korzystające z modemu i złącza RS232.

New Wave... od Hewlett Packarda

O programie tym wspominamy nie tylko ze względu na jego możliwości, ale i z uwagi na oferującą go firmę: oto na rynek oprogramowania dla PC wkracza z własnym pakietem zintegrowanym... Hewlett Packard. Object Management Facility, jądro nowego programu, potrafi natychmiast i bez zawracania głowy użytkownikowi wprowadzić nowe, wpisane z klawiatury dane do tabel, wykresów oraz dokumentów pisanych zawierające dane pobrane z danej tabeli. W odróżnieniu od Dynamic Data Menager Microsoftu New Wave pamięta, skąd pochodziły dane nawet po zakończeniu sesji pracy z dokumentem.

Real UNIX dla 386

Wciąż trwa wyścig producentów systemów operacyjnych dla procesorą 386. Firma Microport Systems oferuje "Real UNIX System V release 3". Czym różni się on od innych implementacji UNIX-a dla 386, trudno powiedzieć, z reklam wynika, że daje pełny dostęp do wszystkich trybów procesorów 386 i 286, natomiast z pewnością można stwierdzić, że od dawna oczekiwane triumfalne wkroczenie UNIX-a w świat mikro staje się faktem: coraz więcej aplikacji dostępnych jest zarówno w wersji DOS, jak i UNIX, a równocześnie coraz więcej programów minikomputerowych pojawia się w wersjach dla świata mikro.

ELI-41

Od dnia, w którym po raz pierwszy ujrzeliśmy SideKick, symulowany prosty kalkulator na ekranie naszego PC uważamy za coś naturalnego i wygodniejszego od podręcznego liczydełka i nawet przestaliśmy się już wstydzić, że tak mądre narzędzie, jak 16-bitowy komputer, wykorzystujemy do prozaicznego udawania kalkulatorka.

Firma Eclipse Logic Inc. uczyniła następny krok w tym kierunku opracowując program ELI-41, symulujący na ekranie PC wyrafinowany tzw. programowany kalkulator naukowy Hewletta Packarda HP-41. Teraz, nie zmieniając przyzwyczajeń związanych z podręcznym kalkulatorem, można będzie programować obliczenia, wykonywać je z 15-cyfrową dokładnością i przechowywać ich wyniki.

Można także - w odróżnieniu od rzeczywistego kalkulatora - zajrzeć do środka tej zabawki i sprawdzić stan rejestrów, flag oraz stosu, a także wydrukować dane i wyniki na drukarce.

ELI-41 może działać niezależnie lub pobierać dane z innych programów. Program korzysta oczywiście z koprocesora 8087/287. Jednym z jego atutów jest dołączana na tej samej dyskietce bogata biblioteka programów użytkowych, jakie przez lata powstały dla HP-41.

Z własnych doświadczeń dawnego użytkownika HP-25 (z 1976 r.) dodam, że wprawny użytkownik po opanowaniu zasad pracy ze stosem i logiki polskiej (reverse polish notation) jest w stanie na tych kalkulatorkach zaprogramować i wykonać skomplikowane obliczenia symulacyjno-statystyczne szybciej niż w jakimkolwiek języku PC, jeśli uwzględnić łączny czas planowania programu, kodowania, uruchamiania i wprowadzania danych.

BASIC Toolbox dla QL

Potwierdza się stara prawda: kultura osiąga dojrzałość w okresach schyłkowych. W 1988 r., w dwa lata po zaprzestaniu produkcji Sinclaira QL, pojawiają się dla tego komputera programy, z których użytkownicy mogą być dumni. Wspomniany tu produkt firmy Compware to kompletne środowisko operacyjne na poziomie bliskim Macintosha.

Term> Tracer

Holenderska firma Ink International oferuje program tłumaczący podstawowe terminy techniczne z zakresu obrotu towarowego i sprzętu komputerowego w językach hiszpańskim, angielskim i francuskim.

opracował W.M.

OGŁOSZENIA EKSPRESOWE

przyjmuje: Krzysztof Karpinski tel. 28 22 01 w. 243 i 290
Biuro Ogłoszeń tel. 28 86 41

TURBO EMULATOR I80 TM
TURBO EMULATOR Z80B TM
TURBO EMULATOR I86/88 TM

PWPT MKS INTECH Sp. z o. o.
44-100 Gliwice skr. pocz. 422
telefon 31 75 31 i 32 01 89

Emulatory układowe mikroprocesorów Zilog Z80, Intel 8080, Intel 8086/8088 współpracujące z komputerami IBM PC/XT/AT przez interfejs szeregowy RS 292C pozwalają na:

- pracę w czasie rzeczywistym
- pracę krokową po cyklu i rozkazie
- zegar wewnętrzny lub zewnętrzny
- czytanie, pisanie, wypełnianie, testowanie, diasemblację pamięci
- obsługę urządzeń wejścia/wyjścia
- oraz wiele innych zleceń

Oferujemy kompletne zestawy do uruchamiania systemów mikrokomputerowych **IBM PC/XT/AT + TURBO EMULATOR xxx + PROGRAMATOR PAMIĘCI EPROM + EPROM ERASER + PROGRAMY ZARZĄDZAJĄCE**

c-51

TANIE I NIEZAWODNE:

IBM-XT+AT-kompatibel

Systemy komputerowe kompatybilne z XT:

ICO 360 **cena 898 DM**
- 8088 CPU, 256K RAM, 8K BIOS-EPROM karta grafiki kolorowej, stacja dysków 360K klawiatura, obudowa IBM

ICO 720 **cena 1099 DM**
- jak ICO 360, lecz z dwoma stacjami dysków

ICO 20 MB **cena 1494 DM**
- jak ICO 360 plus twardy dysk 20 MB

ORAZ:
płyty główne do XT, AT, karta grafiki kolorowej i monochromatycznej, karty EGA, karty sterownika dysków, karty wielofunkcyjnej, itp.

A TAKŻE:
komputery domowe, monitory, drukarki, plotery.

Na wszystkie nasze produkty zapewniamy 12 miesięczną gwarancję, oraz serwis napraw gwarancyjnych i pogwarancyjnych

Systemy komputerowe kompatybilne z AT

ICO AT-1 **cena 1798 DM**
- 80286 CPU, 640K RAM, 64K BIOS-ROM karta grafiki kolorowej, stacja dysków 1.2MB klawiatura, zasilacz 200 W, obudowa IBM

ICO AT 20 **ena 2599 DM**
- jak ICO AT-1, plus twardy dysk 22MB

wyżej wymienione systemy dostępne są także w innych konfiguracjach

OFERUJE FIRMA:

Klaus Jeschke

ADELHEIDSTR 2
6240 KONIGSTEIN (ok. 20 km od Frankfurtu)
tel. 06174/3041 RFN, tlx. 410483
Prowadzimy korespondencję w języku polskim
Serwis prowadzi:
MIKROSERWIS
80-288 GDAŃSK
ul. ORAŃSKA 1A/9 tel. 48 50 63 c-56

STUDIO USŁUG KOMPUTEROWYCH

Samba

BIURO HANDLOWE:
ul. Władysława IV 53
81-384 Gdynia
☎ 21 70 88, 21 95 58
tlx 054660 samba pl

uprzejmie informuje, że prowadzi serwis gwarancyjny i pogwagancyjny wszystkich towarów zakupionych w firmie: KT TECHNOLOGY z Singapuru.

Sprzęt komputerowy najtańszy jaki możecie Państwo spotkać na rynku, zakupić można na miejscu w Singapurze, RFN, Berlinie Zach. lub za pośrednictwem poczty. Przesyłka do Polski "wędruje" tylko DWA TYGODNIE.



Swoim stałym klientom firma KT TECHNOLOGY obiecuje przesłać potwierdzone przez polskie placówki konsularne zaproszenie do odwiedzenia Berlina Zachodniego i Singapuru

Mikrokomputery klasy XT, AT, 386, twarde dyski, drukarki, monitory, plotery i inne urządzenia, a także karty i koprocesory czekają.

DZIŚ DECYZJA - ZA DWA TYGODNIE REALIZACJA.

Tanio, szybko, sprawnie - to dewiza naszych partnerów w Singapurze.

Wszystkie informacje (cenniki, adresy, dane techniczne, prospekty, warunki dostaw otrzymacie Państwo w SAMBIE.

81 - 323 GDYNIA
ul. Wolności 11 tel. 21 95 58

c-52

"CONSULT" - Sp. z o. o. prowadzi serwis gwarancyjny na rzecz firmy:

JONG WAH RADIO TV PTE LTD JET

Informacje techniczno-handlowe, gwarancja w Sp. z o. o. "CONSULT"

GDAŃSK tel. 51 69 21, tlx. 051 24 16

ŁÓDŹ tel. 36 02 04

KRAKÓW tel. 34 19 10

WARSZAWA tel. 41 44 48

c-55

OCZY MASZ JEDNE

najtańsze filtry
filtry ochronne
do monitorów 12,14"
w ciągłej sprzedaży

***poleca TETA** Sp. z o. o.*
Przedsiębiorstwo Innowacyjne
ul. Tenisowa 2 c. obok PRITV
WROCŁAW tel. (0 71) 67 58 25

c-54

UWAGA !!! UWAGA !!!

- **jeśli jesteś posiadaczem telefonu, modemu, oraz komputera klasy PC lub Atari ST, mamy dla Ciebie miłą wiadomość,**
- **w godzinach od 22.00 do 8.00 w dni powszednie czynna jest w gdańsku amatorska, bezpłatna sieć telefoniczna BBS WILDCAT !**
- **nasze paramerty: nr. kierunkowy do Gdańska 058, nasz telefon 52 33 19, transmisja 1200 N 8 1.**
- **po uzyskaniu zgłoszenia connect 1200 naciśnij kilka razy klawisz enter, w ten sposób modem ustali prędkość transmisji,**
- **podając swoją datę urodzenia pisz ją w sposób ciągły miesiąc-dzień-rok, np. 080561,**
- **pytając o Twój numer telefonu, komputer prosi o podanie gò w formacie amerykańskim czyli xxx-xxx-xxxx możesz napisać 123-456 i cztery spacje lub 12 . - 34 .- 56 . . kropka to spacja,**
- **nasza sieć jest jeszcze w stadium prób, twoje telefony pomogą nam ją udoskonalić.**

c-53



Zamierzacie Państwo wprowadzić mikrokomputery do Waszego Zakładu? Wybierzcie właściwego partnera!

Nasza oferta obejmuje:

Produkcję mikrokomputerów ALMA XT/AT
Doradztwo
Instalacje systemów i sieci
Opracowywanie i wdrażanie oprogramowania
Szkolenie
Gwarancje
Serwis pogwarancyjny

WYKONAMY OPROGRAMOWANIE UŻYTKOWE ZGODNE Z PAŃSTWA POTRZEBAMI,
w tym w szczególności systemów płacowych, środków trwałych, finansowo księgowe, magazynowe i wspomagania prac biurowych.

Ko-7

Firma MUEL

oferuje do sprzedaży:

1. **Interfejs do ZX-Spectrum, ZX-Spectrum Plus, Timex 2048,** umożliwiający współpracę z czterema napędami dysków elastycznych, **RAM**-dyskiem, dowolną drukarką graficzną, maszyną do pisania **Robotron S-6120**, monitorem ekranowym, rozszerzający Basic oraz system operacyjny **ZX-Spectrum.**
   Nie zajmuje pamięci RAM !!!
2. Sterowany ikonami programator **Eprom** 2716÷27256 do **ZX Spectrum.**
3. Przeróbkę drukarki DZM 180 na drukarkę graficzną.

Informacja:
tel:33-40-91

Korespondencja:
MUEL ul. Cząstkowska 30, 01-678 Warszawa

Zamówienia:
Spółdzielnia Rzemieślnicza Specjalistyczna Elektryków, ul. Grójecka 128, 02-383 Warszawa

Wykonawca:
MUEL.

Ko-14

dataCo

Przedsiębiorstwo Wdrażania Postępu Technicznego dataCo - Trading

01-710 Warszawa, ul. Włościańska 25
tel. 33-59-73 tlx 816159 datac pl

Posiada w sprzedaży następujące oprogramowanie dla mikrokomputerów kompatybilnych z IBM PC XT/AT:

- instalacja polskich liter (DOS),
- polski edytor tekstowy MS (DOS),
- biblioteka okien dla języka C (DOS, XENIX),
- system finansowo-księgowy (DOS, XENIX),
- system gospodarki materialowej (DOS, XENIX),
- system płacowy (DOS),
- biblioteka graficzna do Turbo Pascala dla karty Hercules (DOS).

Jeżeli jesteś autorem oryginalnego programu aplikacyjnego - skontaktuj się z nami, będziemy pośredniczyć w sprzedaży Twojego programu dbając o ochronę Twoich praw autorskich!

Zapraszamy Naszych Klientów do Działu Oprogramowania Warszawa, ul. Dzika 4 tel. 31-80-75

Ko-71

"MIKROSERWIS"
ul. Marusarzówny 6
80-288 GDAŃSK-MORENA D
Tel.48-50-63 godz.9-17

POLECA NAPRAWY:

- KOMPUTERÓW **SPECTRUM, COMMODORE, AMSTRAD, IBM**
- DRUKAREK **STAR**
- ZASILACZY DO **IBM**
- ORAZ CARTRIDGE DO **C64 (FINAL II, DYSKOBOL) C16, C+4 (UNIWERSAL)**

Ko-122

VIGOR

OFERUJEMY SPRAWDZONĄ NOWOCZESNOŚĆ W ZAKRESIE OPROGRAMOWANIA I SPRZĘTU MIKROKOMPUTEROWEGO UŁATWIAJĄCEGO ZARZĄDZANIE PRZEDSIĘBIORSTWEM. SYSTEMY: GOSPODARKA MATERIAŁOWA, FINANSE I KSIĘGOWOŚĆ, KADRY, PŁACE, KOSZTORYSOWANIE.

SPÓŁKA „VIGOR", 81-450 GDYNIA, ul. REDŁOWSKA 20, TLX 054500 vigor pl, TEL. 22-39-01, 22-32-88

Ko-106

Zakład Elektroniki Komputerowej

ZEKOM

Skr. pocztowa 35, 90-955 Łódź 8, tel. 57-25-83

Użytkownikom komputerów IBM PC/XT/AT/PS2 poleca

TERMINAL EKRANOWY

MT 2583

PC PARTNER



Terminal alfanumeryczny MT 2583 PCP zapewnia zdalny dostęp do zasobów komputera według koncepcji „PC-Shadow".

Struktura obrazu, zestaw znaków, typ klawiatury są zgodne z komputerem IBM PC/AT.

Sposób obsługi terminala – w szczególności korzystanie z oprogramowania – identyczny jak dla komputera.

ZEKOM poleca również kompletne zestawy wieloterminalowe MULTITE przeznaczone do pracy wielostanowiskowej z komputerem klasy IBM PC/XT/AT

NASZE TERMINALE GWARANCJĄ SUKCESU!

Ko-28

Programy komputerowe, instrukcje i udoskonalenia techniczne
pocztą
dla ATARI, AMSTRADA, COMMODORA i IBM
wysyła
Agencja Mikrokomputerowa
Sosnowiec P-157, tel. 63-29-35

Ko-19

Autonomiczny programator 8748/49
oraz
programator pamięci EPROM 2716 ÷ 27512
sprzedam.
Dla instytucji rachunki.
Jacek Kobierzycki
ul. Piwna 39/41 m 9
91-048 Łódź

Ko-125

ELECTRONICS EXPORT

"ELECTRONICS EXPORT" PO.Box 869,London W5,ANGLIA-Tlx 8950511 oneone G (25190001 ref)

Tel (0-0441) 993 7000 - **Showroom i sklep** ; 19,Queens Parade,London W5,Ealing

ATARI ST

Do każdego ST gratis "Starter pack" wartości 75 edytor tekstów, 5 dysków z programami i 5 dysków czystych, wszystkie ST z myszką.

520 STFM + mon.mono SM125 **(NOWOŚĆ)**	**400**
520 STM + DRIVE SF354 + mon.SM125	360
520 STM + DRIVE SF314	335
520 STM + DRIVE SF 314 + mon.SM125	425
520 STFM (wbudowany drive)	280
1040 STF (wbudowany drive)	450
1040 STF + mon.SM125	540
MEGA 2MB **(NOWOŚĆ)** DRIVE 720 K	**810**
MEGA 2MB mon.SM125	900
DRIVE SF354 (360K)	135
DRIVE SF314 (720) obniżka	145
DRIVE 5,25" "CUMANA" 720K	170
Dysk twardy 20MB SH204 (lub SUPRA)	499
Monitor mono SM125	135
Drukarka laserowa ATARI SLM804	1150
Monitor kol."PHILIPS"8833 (600x285)	245
Modulator TV do 1040 STM (wbudowany)	45

Profesjonalny softwere "TIMEWORKS" mogą być łączone między sobą.

DATA MANAGER *(database/graphics)*	35
SWIFT CALC *(spreadsheet)*	35
WORD WRITER *(edytor tekstu)*	70
TIMEWORKS DESKTOP PUBLISHER	90

DESKTOP PUBLISHING

Komplet obejmujący wszystko do stworzenia DTP
MEGA 2MB, mon. mono SM125, drukarka laserowa ATARI SLM 804,oprogramowanie po polsku i angielsku, 20 dysków 3,5"DSDD — 2170

DRUKARKI CITIZEN

120D (NLQ, 120 Z/SEK, 25 cm. papier zwykły, komputerowy)	145
MSP 15E (NLQ, 160 Z/SEK, 40 cm. papier zwykły, komputerowy)	250
HQP 40 (24 igły, 200 Z/SEK, 40 cm. papier zwykły, komputerowy. 42 K bufor, Epson LQ800 COMPAT, możliwość druku kolorowego)	360
Drukarka laserowa OVERTURE 110 (10 str/min, 300 z/cal, 512 kRAM, format A4)	3190

DRUKARKI STAR roczna gwarancja

SG 10, NLQ, 120 Z/SEK, 25 cm. taśma	180
LC 10, NLQ, 120 Z/SEK, 25 cm, **NOWOŚĆ**	180
GEMINI 15X, 120 Z/SEK, 40 cm, taśma	160
NX 15, NLQ, 120 Z/SEK, 40 cm, kaseta	300
NB 24-10, 24-igłowa, 216 Z/SEK, 25 cm	445
NB 24-15, 24-igłowa, 216 Z/SEK, 40 cm	565
NB 15, 24-igłowa, 300 Z/SEK, 40 cm	635
ND 10, NLQ, 180 Z/SEK	285
ND 15, NLQ, 180 Z/SEK	380
NR 10, NLQ, 240 Z/SEK	350
NR 15, NLQ, 240 Z/SEK	460
Kabel drukarka-IBM/ST	10

PLOTERY A3

HITACHI 672 XD (nowy model)	499
ROLAND DXY 880A	650

DYSKI: NASCHUA, 3M - za 10, szt.

5,25" 10£ powyżej 100 szt. 6£
5,25" DSHD/AT 20£, powyżej 50 szt. 14£
3,5" SSDD 15£, powyżej 100 szt. 11£
3,5" DSDD 20£, powyżej 50 szt. 14£
3" "AMSOFT" 25£, powyżej 50 szt. 22£

OPUS PC III/XT TURBO 10 MHz

1 MB RAM, monitor bursztynowy 14"
Za dodatkową opłatą 50£, do systemu 4, 5 lub 8 otrzymasz drukarkę STAR SG 10 lub GEMINI 15 X.

System 2, 1 stacja dysków 360 K	599
System 3, 2 stacje dysków 360 K	649
System 3 PLUS, 1 x 360, 1 x 1.2 MB	699
System 4, 1 x 360, dysk twardy 30 MB	949
System 5, 2 x 360, dysk twardy 30 MB	999
System 5 PLUS, 1 x 360 K, 1.2 MB, 30 MB	1049

OPUS PCV/AT TURBO 10 MHz

1 MB RAM, monitor bursztynowy 14"
System 8, 1 x 360 K, 1.2 MB, dysk twardy 30 MB — **1299**
Dopłata do systemu EGA-KOLOR — 300

VIP XT/AT Niższe ceny od 575 USD

NOWOŚĆ, popularne komputery sprowadzane bezpośrednio z Tajwanu do Polski. Najniższe ceny w dolarach amerykańskich obejmują także koszty frachtu, przeglądu zerowego i gwarancji.

VIP XT/SD TURBO

(4,77 MHz/10 MHz, 256 K RAM. max. możliwe 640 K RAM + 384 K RAM dysk, 8 gniazd, 1 drive 360 K-Chinon/Matsushita. HERKULES. klawiatura 84 klawisze. CENTRONICS port, 150 W zasilanie, obudowa BABY) Dos, system operacyjny (dysk) — **625 USD**

VIP XT/TF TURBO

(tak jak VIP XT/SD + drugi drive 360 K, 640 K RAM. 128 K RAM dysk) — **725 USD**

VIP XT/HD TURBO

Tak jak "VIP XT/SD" + twardy dysk 20MB Seagate ST-225 — **999** USD

VIP AT

10 MHz (1 MB RAM, 6/10 MHz ZERO WAIT. 8/12 ONE WAIT procesor 80286-10, może używać koprocesor 80287-10 8 gniazd. 1 drive 360, 1.2 MB - Chinon/NEC, dysk twardy SEAGATE 20 MB, karta HERKULES/ CENTR/RS 232, klawiatura 102 klawisze. obudowa BABY. 180 W. — **1550** USD

MONITOR MONO amber 12"	150 USD
MONITOR KOLOR 14" + karta EGA	599 USD

Po dokonaniu wpłaty w funtach angielskich (tylko VIP w USD) zamówienie wraz z kopią wpłaty wysłać do nas listem poleconym. Przesyłki wysyłamy drogą lotniczą do Warszawy. Do sumy zamówienia dolicz 5£. Koszty frachtu opłaca odbiorca w złotych przy odbiorze.

UWAGA! Cena komputera VIP obejmuje wszystkie koszty. Na żądanie artykuły poniżej 150£ możemy wysłać samochodem do Warszawy. 8£ pokrywa koszty całego frachtu. Zamówienie min. 30£ na jeden adres. Komputery OPUS PC, VIP, ATARI ST objęte są przeglądem zerowym i roczną gwarancją przez autoryzowany serwis firmy **UNICOMP** tel. w Warszawie 55-45-54

Jeżeli zaszły jakieś zmiany w cenniku - patrz rubryka *"Na 10 dni przed drukiem"*.

NASZ BANK: BANK HANDLOWY SA w Warszawie, oddział London, 4 Coleman str., LONDON EC2, ANGLIA, No-konta 20 00 47-001

STUDIO USŁUG KOMPUTEROWYCH sp. z o.o.

Samba

BIURO HANDLOWE:
ul. Władysława IV 53/3
81-384 Gdynia
☎21 70 88, 21 95 58

SZANOWNI PAŃSTWO!

ZARZĄDZANIE WSPÓŁCZESNYM PRZEDSIĘBIORSTWEM WYMAGA PODSTAWOWYCH NARZĘDZI, JAKIMI SĄ

KOMPUTERY I ICH OPROGRAMOWANIE

Nasze Studio Usług Komputerowych oferuje Państwu pomoc w:

KOMPLEKSOWEJ KOMPUTERYZACJI CAŁEGO PRZEDSIĘBIORSTWA

Szczegółowe informacje uzyskają Państwo w naszym Biurze Handlowym oraz w filiach:

91-135 Łódź	75-530 Koszalin	50-052 Wrocław	31-033 Kraków	65-064 Zielona Góra
ul. Grabieniec 12/20	ul. M. Buczka 4	ul. Widok 5/16	Dom Turysty	ul. Kościelna 1
tel. 52-69-12	tel. 503-31	tel. 44-81-64	ul. Westerplatte 15/16	tel. 722-88
			tel. 22-96-65 wew. 197	

UŁATWIAMY ZARZĄDZANIE

Ko-83

refleks

NASZA OFERTA!!!

PWPO-T „REFLEKS" Sp. z o.o. informuje,

że prowadzi działalność serwisową na rzecz firmy ASCOM TECHNOLOGIES (FAR EAST) PTE LTD z SINGAPURU. Sprzęt zakupiony w firmie ASCOM podlega bezpłatnej rocznej gwarancji, w czasie której funkcje gwaranta sprawuje na zasadzie wyłączności PWPOT REFLEKS.

Zakupiony wysyłkowo lub osobiście w firmie ASCOM sprzęt:

- kompletne zestawy mikrokomputerów PC/XT 6/8/10 MHz, PC/AT 8/10/12 MHz, PC/386 12/16/20 MHz.
- pełny asortyment kart CSKD, wyposażenia i akcesoriów umożliwiających samodzielne zbudowanie mikrokomputera lub rozszerzenie zestawu już posiadanego (karty główne, grafiki, kontrolery, karty obsługi wejść/wyjść, kable, obudowy, klawiatury, zasilacze).
- pełny asortyment urządzeń zewnętrznych, takich jak: monitory monochromatyczne i kolorowe (szeroka gama typów o różnej rozdzielczości), pamięci na miękkich dyskach i napędy dysków twardych (o bardzo dużej pojemności i krótkim czasie dostępu), różne typy ploterów i digitizerów jest testowany i sprawdzany bezpłatnie w Zakładzie Serwisowym REFLEKS - Raszyn, ul. Mickiewicza 5A.

UŻYTKOWNIK OTRZYMUJE TYLKO SPRZĘT SPRAWNY I WYSOKIEJ JAKOŚCI!

Ponadto REFLEKS Sp. z o.o. udzieli Państwu odpłatnie dodatkowych informacji technicznych i doradztwa w sprawach handlowych:

1. Dział Handlowy, 02-051 Warszawa, ul. Glogera 1, tel. 02/659-20-41
2. Zakład Serwisowy - Raszyn, ul. Mickiewicza 5A
3. Sklep SPHW nr 509 - Studio Komputerowe REFLEKSU, ul. Prosta 2/14, tel. 24-01-48

Przedsiębiorstwo Wdrażania Postępu Organizacyjno-Technicznego refleks **Sp. z o.o.**

Ko-22

Kolgar

Bromenrijk 31
1112 El Diemen,
HOLANDIA





Modele desktop, portable, pionowo stojące

Model XT

* 4,77/10 MHz, 640 KB RAM, integracja sterowników: CGA, HERCULES i Multi I/O na karcie głównej.
* Dwa napędy dysków 360 KB RAM.
* Klawiatura 84 klawisze.
* Monitor monochromatyczny 12".
* Zasilacz 150 W

Model AT

* 8/12 MHz, 640 KB RAM, karta CGA + Hercules + RS 232 + Centronics, karta kontrolerów FDD/HDD.
* Jeden napęd dysków 1,2 MB.
* Dysk twardy 20 MB.
* Klawiatura 84 klawisze.
* Monitor monochromatyczny 12".
* Zasilacz 200 W.

- Bogata gama konfiguracji i wyposażenia dodatkowego
- Transport wliczony w cenę komputera.
- Rok gwarancji, serwis dostępny w Polsce.
- Do zakupionych komputerów dołączamy trzy programy.
- Bardzo atrakcyjne ceny w granicach

599 ÷ 2700 USD.



Model PC XT

* 4,77/10 MHz, 256 KB RAM, płyta główna zintegrowana Hercules + CGA + pełne Multi I/O
* Napęd dysków 360 KB.
* Klawiatura 84 klawisze.
* Monitor 12".
* Zasilacz 150 W.
* **CENA 599 USD.**

- **Handy Scanner do IBM** **315 USD.**
- **Drukarka inkjet** **1950 USD.**

Telefon w Holandii (mówimy po polsku):
0-031-20-95 20 33, w godz. 10-13.

Telefon w Warszawie czynny od pon. do pt.
47-45-81, w godz. 14-18.

C-8

Dotychczas pojawiające się w tej rubryce ceny z zagranicy pochodziły z witryn sklepowych lub ogłoszeń. Tym razem zapraszam na prawdziwą giełdę: Boston Computer Exchange (BoCoEx) wraz z jej 82 biurami regionalnymi jest największym w USA pośrednikiem w handlu używanymi komputerami, zawierającym codziennie setki transakcji. Jej szef i założyciel, Alexander Randall, deklaruje: "rynek używanych komputerów rozwija się zawsze szczególnie dynamicznie po wprowadzeniu do obrotu nowego, udanego modelu. Przełomowe momenty w rozwoju BoCoEx to pojawienie się IBM AT, co zdynamizowało wtórny obrót IBM PC/XT oraz Maca Plus, co stworzyło rynek wczesnych modeli Macintosha".

BoCoEx publikuje dwa razy w tygodniu w największym bostońskim dzienniku - Boston Globe - cennik oferowanego sprzętu, codziennie natomiast wysyła do zainteresowanych kupnem wykaz komputerów, jakimi w danym momencie dysponuje. Wykaz ten można zamówić listownie (box 1177, Boston, MA 02103), telefonicznie lub modemem, korzystając z bostońskiego serwisu informacyjnego CitiNet. Sprzęt przyjęty do sprzedania po dokładnym przeglądzie otrzymuje 30-dniową gwarancję. Transakcje dokonywane są w ciągu trzech dni od złożenia przez potencjalnego nabywcę zamówienia (popartego czekiem lub numerem karty kredytowej). Koszty dostarczenia mu komputera ponosi BoCoEx. Kupujący w ciągu 48 godzin może sprawdzić i wypróbować zamówiony sprzęt, a gdy nie jest z niego zadowolony - zrezygnować, nie ponosząc żadnych kosztów. Po sfinalizowaniu transakcji Boston Computer Exchange przekazuje sprzedającemu 90% uzyskanej ze sprzedaży kwoty.

Udział BoCoEx na rynku sprzętu używanego stale rośnie. Nabywcy korzystają z usług firmy, gdyż sprzęt w BoCoEx jest tańszy niż w ofertach zamieszczanych w lokalnej prasie przez dotychczasowych użytkowników i sprawdzony - pośrednictwo BoCoEx ogranicza ryzyko "kupowania kota w worku". Sprzedawcy cenią sobie szybkość działania firmy.

Głównymi konkurentami BoCoEx są podobnie działające firmy Micro Xchange z Santa Barbara i Used Computer Store z Berkeley w Kalifornii, dające - przy nieznacznie wyższej prowizji - 90-dniową gwarancję na zakupiony sprzęt, wliczoną w jego cenę.

Nabywcami komputerów IBM PC są głównie osoby korzystające z tego rodzaju sprzętu w pracy - nie opłaca się im poznawać w domu innego sprzętu, szukają raczej szansy na pracę w domu. "Przesiadają się" na IBM najczęściej z komputerów Apple II. Te z kolei kupują rodzice dla swych pociech, które znając ten sprzęt z zajęć w szkole bez kłopotu mogą je uruchomić i do znudzenia ćwiczyć w domu kierowanie odrzutowcem, rozwiązywanie scrabble czy montowanie rodzinnej gazetki.

Stosunkowo niewielkim zainteresowaniem cieszą się komputery Commodore 64 i Atari 130 XE. Ich właścicielom myślącym o sprzęcie wyższej klasy w sukurs przychodzi firma Protecto, która w zamian za używanego Commodore 64 ze stacją dysków oferuje 200-dolarową zniżkę przy kupnie Lasera PC (nowego). Jest to interesująca propozycja, gdyż sprzedając używany komputer hurtownikowi (a więc np. firmie wysyłkowej) nie można liczyć na uzyskanie wysokiej ceny - co łatwo sprawdzić w "Orion Computer Blue Book", wydawanym w Colorado przez Rodgera Rohrsa przewodniku po cenach używanych komputerów. Oto przykładowe proporcje cen z wydania z 1987 r.:

komputer	cena nowego sprzętu (sugerowana przez wytwórcę)	średnia cena skupu sprzętu używanego	średnia cena detaliczna sprzętu używanego
Apple IIc mono	$ 995	$ 126	$ 297
Apple Mac Plus	2195	446	1150
Atari 520 ST mono	795	115	310
Atari 1040 ST color	1099	198	475
Commodore C 128	495	30	125
Commodore Amiga color	1295	211	513
Compaq Portable	2199	291	710

W BoCoEx używany sprzęt jest średnio od 30 do 50% tańszy od nowego. Cena zależy od stanu technicznego, popularności danego modelu (dostępność oprogramowania!) oraz sezonu. Sprzedający często dołącza gratis swój zbiór programów i podręczników.

Używane komputery najtańsze są w styczniu - widać wówczas wzmożoną falę pozbywania się starych komputerów po masowych zakupach najnowszych modeli na Gwiazdkę. Wyraźny wzrost cen jest natomiast widoczny w kwietniu - w okresie składania zeznań podatkowych, na początku roku szkolnego i przed świętami Bożego Narodzenia.

Poniższe ceny BoCoEx z 22 kwietnia 1988 r. są więc o kilka procent wyższe niż w lutym i marcu:

IBM PC 076	$ 825
IBM XT 086	1225
IBM XT 089	1650
IBM AT 099	2350
IBM AT 239	2650
IBM AT 339	3650
IBM PS/2 model 50	2475
IBM PS/2 model 60	2650
Compaq Portable I	825
Compaq Port. II M4	2100
Comp. Por. III 40M	3200
Comp. Plus	1175
Comp. Deskpro 20M	1400
Comp. Dskpr. 286 M2	2425
Comp. Dskpr. 386 M4	4075
Macintosh 512	725
Macintosh 512e	850
Macintosh Plus	1150
Macintosh SE	1925
Macintosh SE 20M	2375
Macintosh II	3900
Apple II GS	850
Apple IIe	725
Apple IIc	650
Apple Laserwriter	3100
HP Laserjet	925
NEC Multispeed EL	900

DANUTA MAJEWSKA

